

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra kybernetiky a biomedicínského inženýrství

**Vestavěný řídicí systém pro analýzu
základních slov řečového signálu**

**Embedded System for Speech Word
Recognition**

Zadání diplomové práce

Student: **Bc. Matěj Kaloč**

Studijní program: N2649 Elektrotechnika

Studijní obor: 2612T041 Řídicí a informační systémy

Téma: Vestavěný řídicí systém pro analýzu základních slov řečového signálu
Embedded System for Speech Word Recognition

Jazyk vypracování: čeština

Zásady pro vypracování:

1. Rozbor problematiky zvukové detekce řečového signálu s popisem metod.
2. Návrh systému pro detekci slovních spojení v hlasovém projevu.
3. Realizace systému pro detekci slovních spojení v hlasovém projevu.
4. Měření a analýza s vyhodnocením skutečných parametrů systému.
5. Zhodnocení dosažených výsledků závěrečné práce.

Seznam doporučené odborné literatury:

- [1] JINYU, Li, Li DENG, Reinhold HAEB-UMBACH a Yifan GONG. *Robust Automatic Speech Recognition*. Oxford, UK: Academic Press, 2015. ISBN 978-0128023983.
- [2] PSUTKA, Josef. *Mluvíme s počítačem česky*. Praha: Academia, 2006. ISBN 80-200-1309-1.
- [3] SIGMUND, Milan. *Zpracování řečových signálů*. Brno: VUT v Brně, 2006. Učební text pro FEKT. ISSN 80-214-3300-0.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. Zdeněk Macháček, Ph.D.**

Datum zadání: 01.09.2017

Datum odevzdání: 30.04.2019



doc. Ing. Jiří Koziorek, Ph.D.
vedoucí katedry



prof. Ing. Pavel Brandštetter, CSc.
děkan fakulty

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární
prameny a publikace, ze kterých jsem čerpal.

V Ostravě 30. dubna 2019

.....


Souhlasím se zveřejněním této diplomové práce dle požadavků čl. 26, odst. 9 Studijního a zkušebního řádu pro studium v magisterských programech VŠB-TU Ostrava.

V Ostravě 30. dubna 2019


.....

Rád bych tímto poděkoval svému vedoucímu práce za pomoc, kterou mi při sepisování práce poskytl a také všem, kteří mě při psaní práce podporovali a všem, kdo se účastnili testování a pomohli mi tak verifikovat výsledky.

Abstrakt

Práce zkoumá problematiku detekce klíčového slova a s ní související problémy jako je výběr mikrofonu a AD převodníku. V práci jsou také popsány dostupné nástroje na detekci slova. V rámci práce byl navrhnout systém dvou jednoúčelových zařízení, z nichž jedno se stará o detekci slova a druhé o reakci na detekované slovo. Detekce klíčového slova probíhá pomocí knihovny Snowboy na jednodeskovém počítači Raspberry Pi 3. Navržený systém byl také realizován a verifikován. Ve verifikaci se nachází výsledky testu kvality detekce slova tímto zařízením.

Klíčová slova: Detekce slova, Snowboy, Arduino, Raspberry Pi, ReSpeaker

Abstract

This Thesis examines the issue of word spotting in speech and related tasks such as choosing a microphone type or the selection of AD converter. The components which make word spotting possible are described in this thesis. Within the framework of this thesis, a system of two dedicated devices was designed. One of the devices tries to detect words in speech and another reacts to detected words. The Snowboy library on Raspberry Pi 3 is responsible for detecting words. The System has been implemented and verified. The results of detection quality tests are located in the verification section.

Key Words: Word spotting, Snowboy, Arduino, Raspberry Pi, ReSpeaker

Obsah

Seznam použitých zkratk a symbolů	9
Seznam obrázků	10
Seznam tabulek	12
1 Úvod	13
2 Teoretický rozbor	15
2.1 Metody převodu zvuku na elektrický signál	15
2.2 Metody digitalizace signálu	17
2.3 Metody rozpoznání řeči	19
2.4 Snowboy	21
2.5 Google asistent	22
2.6 Alexa voice service	23
2.7 Sopare	23
3 Návrh vestavěného řídicího systému pro analýzu základních slov řečového signálu	25
3.1 Návrh hlasového ovladače	26
3.2 Návrh Spínače	26
3.3 Návrh uživatelského programu	26
4 Realizace vestavěného řídicího systému pro analýzu základních slov řečového signálu	28
4.1 Hardware Raspberry Pi	28
4.2 Hardware ReSpeaker	29
4.3 Hardware Zesilovače	30
4.4 Propojení RPI3, ReSpeakeru a Zesilovače	31
4.5 Software Hlasového ovladače	33
4.6 Hardware Spínače	36
4.7 Software Spínače	40
4.8 PC Program	45
5 Verifikace	51
5.1 Forma verifikace	51
5.2 Naměřená data	52
5.3 Zhodnocení výsledků měření	61

6 Závěr	65
Literatura	66
7 Seznam příloh	72
Přílohy	73
A CD	73

Seznam použitých zkratek a symbolů

LED	– Light Emitting Diode
USB	– Universal Serial Bus
HDMI	– High-Definition Multi-media Interface
GPIO	– General-Purpose Input/Output
LED	– Light-Emitting Diode
I2C	– Inter-Integrated Circuit
ADC	– Analog-to-Digital Converter
LSB	– Least Significant Bit
RPI3	– Raspberry Pi 3
OZ	– Operační Zesilovač
LAN	– Local Area Network
HW	– Hardware
DHCP	– Dynamic Host Configuration Protocol
ICMP	– Internet Control Message Protocol
ASCII	– American Standard Code for Information Interchange
EEPROM	– Electrically Erasable Programmable Read-Only Memory
MEMS	– MikroElektroMechanické Systémy
ACK	– Alexa Connect Kit
IFTTT	– If This Then That
DPS	– Deska Plošných Spojů
DC	– Stejnoseměrný Proud
TDNN	– Time Delay Neural Network

Seznam obrázků

1	Příklad MEMS mikrofonu	17
2	Schéma převodníku sigma delta	19
3	Schéma propojení částí systému	25
4	Návrh propojení modulů	26
5	Návrh užití uživatelského programu	27
6	Transformace informace systémem	28
7	Změny barev tří RGB LED při běhu programu	30
8	Schéma propojení modulů a Raspberry pi	31
9	Propojení zesilovače, ReSpeaker a Raspberry pi	32
10	Model krabičky Hlasového ovladače	32
11	Diagram startovací sekvence programu a rozpoznávání klíčových slov	34
12	Diagram popisující vlákno myPcCommunicationThread	35
13	Model DPS Spínače s krabičkou na DIN lištu	37
14	Schéma zapojení mikrokontroléru Atmega328 ve Spínači	37
15	Schéma zapojení převodníku CH340G ve Spínači	38
16	Schéma zapojení Ethernetového rozhraní ve Spínači	39
17	Schéma zapojení DC/DC konvertoru ve Spínači	39
18	Schéma zapojení výstupního relé	40
19	Blokové schéma životního cyklu programu Spínače	41
20	Diagram užití programu na konfiguraci Hlasového ovladače a Spínače	45
21	Pravá stromová struktura nazvaná jako seznam klíčových slov a levá stromová struktura nazvaná jako seznam Spínačů	46
22	Spojení se Spínačem	46
23	Pole pro konfiguraci Spínače	47
24	Tlačítka pro uložení a načtení konfigurace	47
25	Editace konfigurace	48
26	Pole pro navázání komunikace s Hlasovým ovladačem	48
27	Pole pro nahrávání klíčových slov	49
28	Pole pro vytváření modelů	49
29	Pole pro uložení modelu	50
30	Úspěšnost správné detekce napoprvé při různých vzdálenostech a okolním hluku .	62
31	Znázornění maximálního a minimálního počtu pokusů, hodnocy prvního a třetího kvartilu a mediánu při různých vzdálenostech a okolním hluku u žen	62
32	Znázornění maximálního a minimálního počtu pokusů, hodnocy prvního a třetího kvartilu a mediánu při různých vzdálenostech a okolním hluku u mužů	62
33	Znázornění chybovosti detekce při různých vzdálenostech a okolním hluku	63
34	Znázornění úspěšnosti detekce jednotlivých řečníků při použití cizího modelu . .	63

35	Znázornění chybovosti detekce jednotlivých řečníků při použití cizího modelu . .	64
----	--	----

Seznam tabulek

1	Porovnání rychlosti a rozlišení ADC	18
2	Data uložená v paměti EEPROM	42
3	Význam bitů v prvním bytu příkazu Spínači	43
4	Formát sériové komunikace	44
5	Naměřená data řečníkem Gabriela ml. testu správnosti detekce při použití modelů řečníka	53
6	Naměřená data řečníkem Gabriela st. testu správnosti detekce při použití modelů řečníka	54
7	Naměřená data řečníkem Hana testu správnosti detekce při použití modelů řečníka	54
8	Naměřená data řečníkem Iveta testu správnosti detekce při použití modelů řečníka	55
9	Naměřená data řečníkem Jiří ml. testu správnosti detekce při použití modelů řečníka	56
10	Naměřená data řečníkem Jiří st. testu správnosti detekce při použití modelů řečníka	56
11	Naměřená data řečníkem Karel testu správnosti detekce při použití modelů řečníka	57
12	Naměřená data řečníkem Karolína testu správnosti detekce při použití modelů řečníka	57
13	Naměřená data řečníkem Vladislav testu správnosti detekce při použití modelů řečníka	58
14	Naměřená data řečníkem Matěj testu správnosti detekce při použití modelů řečníka	58
15	Naměřená data řečníkem Michal testu správnosti detekce při použití modelů řečníka	59
16	Naměřená data řečníkem Sarah testu správnosti detekce při použití modelů řečníka	59
17	Naměřená data řečníkem Viktor testu správnosti detekce při použití modelů řečníka	60
18	Naměřená data všech řečníků testu správnosti detekce při použití modelů cizího řečníka (Matěj) ve sloupcích 2 až 13. Pro srovnání stejný test řečníka Matěj oproti vlastnímu modelu ve sloupci 14	61

1 Úvod

V teoretickém rozboru jsou popsány všechny kroky, které je potřeba udělat pro detekci slova v mluveném projevu. Kapitola začíná definicí zvuku, hlasu a popisem souvisejících fyzikálních veličin. Následuje popis mikrofونů běžně používaných pro převod zvuku z akustického vlnění na elektrické. Kapitola mikrofony nejen popisuje ale také krátce vysvětluje jejich fyzikální princip. Typy mikrofونů, jimž se práce věnuje, jsou kondenzátorový mikrofون, dynamický mikrofون a MEMS mikrofون.

V následujících odstavcích je vysvětleno, co to je digitalizace signálu a jsou popsány metody digitalizace signálu. Kapitola se stručně věnuje pojmům vzorkování a kvantování. Následuje popis a vysvětlení fyzikálního principu nejběžnějších typů ADC a také vzájemné porovnání jejich základních vlastností. Popis převodníků je často vztažen k jejich použitelnosti pro převod hlasu.

V podkapitole Metody rozpoznání řeči je vysvětlen význam pojmu rozpoznání řeči a jsou popsány různé typy a dělení rozpoznání řeči. Jednotlivé typy rozpoznání řeči jsou porovnány z hlediska náročnosti na výpočetní výkon a jejich složitosti. Kapitola také popisuje způsoby detekce slova. V posledních kapitolách teoretického rozboru jsou popsány nástroje sloužící k detekci slov a rozdíly mezi těmito nástroji. Kapitoly se věnují nástrojům Snowboy, Google asistent, Alexa voice service a Sopare.

Kapitola Návrh začíná jako popis navrženého systému na detekci slov. V kapitole jsou popsány jednotlivé části systému a propojení těchto částí. Návrh jednotlivých částí systému je poté blíže popsán v samostatných podkapitolách. V kapitolách jsou popsány požadavky na jednotlivé části systému a jejich návrh.

Kapitola Realizace začíná obecným popisem použitého hardwaru a softwaru. Použitý hardware a software je poté popsán v samostatných podkapitolách. První podkapitolou je Hardware Raspberry Pi, v této kapitole jsou vyzdvíženy pro práci nejdůležitější části tohoto jednodeskového počítače. Podobně je také popsán hardware modulu ReSpeaker a modulu Zesilovače a připojení mezi těmito moduly a RPI3. Následující podkapitola popisuje fungování programu v tomto modulu. Podkapitola je rozdělena dle úkolů, které tento program vykonává. Kapitola se věnuje komunikaci programu v modulu s uživatelským programem, vyhledání reléových modulů v síti, reakci na detekovaná slova a komunikaci s reléovými moduly.

Následuje podkapitola Hardware Spínače, kde je popsáno fungování reléového modulu. V kapitole jsou popsány všechny části tohoto modulu. Za touto podkapitolou následuje popis softwaru reléového modulu. V kapitole jsou popsány všechny úkoly, které program musí vykonat a rozepsány použité komunikační protokoly.

V poslední části kapitoly Návrh je popis uživatelského počítačového programu. Podkapitola popisuje funkce programu, konfiguraci modulů pomocí programu a vytváření klíčových slov.

Poslední kapitolou je Verifikace. V této kapitole je nejprve popsána forma verifikace systému a průběh testů. Následuje výpis naměřených dat s krátkým popisem. V poslední části kapitoly

je zhodnocení výsledků měření.

2 Teoretický rozbor

V kapitole 2.1 teoretického rozboru jsou popsány metody digitalizace zvuku. Kapitola stručně popisuje fyzikální vlastnosti zvuku, definuje, co je to hlas a vysvětluje vznik hlasu.

V následujících odstavcích kapitoly jsou popsány kroky, které je potřeba udělat pro digitalizaci zvuku. Zvuk je nejprve potřeba převést z akustického tlaku na elektrický signál pomocí mikrofону. Kapitola tedy popisuje nejběžnější typy mikrofónů. Zaměřuje se při tom na mikrofony vhodnější na snímání lidského hlasu.

Další kapitola je kapitola 2.2. V této kapitole je vysvětleno, co je to digitalizace signálu a její spojitost se zvukem. Kapitola také popisuje jednotlivé části digitalizace, tedy vzorkování a kvantování. Poslední odstavce kapitoly popisují běžné typy ADC s ohledem na použitelnost při digitalizaci lidského hlasu.

Kapitola 2.3 se věnuje metodám zpracování řeči. Popisuje rozdíly mezi různými typy systémů na zpracování řeči a jejich vzájemné výhody a nevýhody. V kapitole jsou také popsány běžně používané metody na detekci slov.

V kapitole 2.4 je popsán nástroj Snowboy. Kapitola popisuje výhody a nevýhody tohoto nástroje oproti jiným běžně používaným nástrojům a také se věnuje způsobům použití tohoto nástroje.

V kapitolách 2.5, 2.6 a 2.7 jsou popsány podobné nástroje jako je Snowboy obdobným způsobem. Kapitoly popisují možnosti použití, výhody, nevýhody a limitace těchto nástrojů.

2.1 Metody převodu zvuku na elektrický signál

Věda, která zkoumá vznik, šíření a vnímání zvuku se nazývá akustika. Zvuk je mechanické vlnění v elastickém prostředí, které je schopno vyvolat sluchový vjem. Člověk je schopen vnímat vlnění o frekvenci přibližně 16 Hz až 20 kHz. I když mechanické vlnění mimo tento frekvenční rozsah zvukový vjem nevyvolává, je někdy také označováno jako zvuk. Lidské ucho je nejcitlivější na frekvenční rozsah 2 až 4 kHz. Tento frekvenční rozsah je nejdůležitější pro srozumitelnost řeči. Největší informační hodnotu má zvuk ve frekvenčním pásmu 0,5 až 2 kHz. [6] [7]

Základní fyzikální veličiny při zpracování zvuku jsou frekvence a intenzita. Frekvence udává počet kmitů zvukové vlny za jednu sekundu. Jednotkou je Herz. Intenzita zvuku je definována jako akustický výkon na jednotku plochy. Hladina intenzity zvuku je veličina, která udává intenzitu zvuku v jednotkách decibel. [6][7] [10]

Zvuk vytvářený hlasivkami se nazývá hlas. Zdrojem hlasu jsou hlasivky. V hlasivkách vzniká hranatý tón modulováním proudu vzduchu. Tento hranatý tón je modifikován v dutinách vokálního traktu. Modifikovaný zvuk vychází do prostoru z úst, případně z nosu. Zvukovou stránku jazyka zkoumá fonetika. [10] [9]

Před zpracováním akustického zvukového signálu je potřeba ho nejprve převést na analogový elektrický signál. Po převodu je elektrický signál obvykle zesílen a digitalizován. K převodu zvuku

na elektrický signál slouží mikrofon. Momentálně se k tomuto účelu běžně používají 3 základní druhy mikrofonů. Kondenzátorový, dynamický a MEMS mikrofon. [11] [12]

2.1.1 Kondenzátorový mikrofon

V kondenzátorovém mikrofonu je membrána, kterou rozkmitávají akustické kmity, neboli zvuk. Tato membrána je jednou z elektrod kondenzátoru. Při rozkmitání membrány se mění vzájemná vzdálenost elektrod a s ní i kapacita kondenzátoru. Kapacita kondenzátoru může být u mikrofonů měřena různými metodami. [13] [11] [12]

Mikrofony se stejnosměrným offsetem měří kapacitu kondenzátoru tak, že kondenzátor mikrofonu nabijí na konstantní náboj z měkkého zdroje napětí. Napětí z měnící se kapacity mikrofonu je získáno pomocí rovnice pro výpočet kapacity 1, kde C je kapacita ve faradech, Q je náboj v coulumbech a U je napětí ve voltech. Díky měkkému zdroji napětí může být náboj považován za konstantu. Změna kapacity se tedy projeví pouze změnou napětí. Kmitání membrány, která je elektrodou kondenzátoru, se projeví jako kolísání napětí okolo offsetu. Signál poté projde nízkofrekvenčním filtrem, ve kterém je odstraněn offset. Poté je signál zesílen zesilovačem.[13] [11] [12]

$$C = \frac{Q}{U} \quad (1)$$

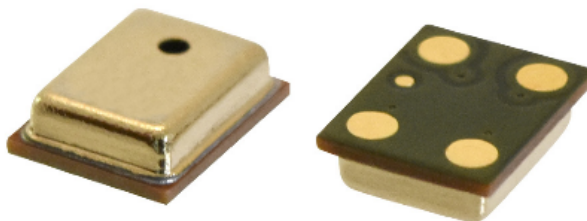
Dalším typem kondenzátorového mikrofonu je mikrofon ve vysokofrekvenčním zapojení. Kondenzátor mikrofonu je součástí vysokofrekvenčního oscilátoru. Změna kapacity kondenzátoru způsobí změnu frekvence oscilátoru. Zvukový signál je získán demodulací vysokofrekvenčního signálu.[13] [11] [12]

Třetím typem kondenzátorového mikrofonu je elektretový mikrofon. Elektretové mikrofony fungují obdobně jako mikrofony se stejnosměrným offsetem s tím rozdílem, že náboj dodávaný z externího zdroje je nahrazen permanentním nábojem v elektretu. Zbylé části procesu získávání napěťového signálu ze zvukového jsou obdobné. [13] [11] [12] [14]

2.1.2 Dynamický mikrofon

Dynamické mikrofony fungují na principu pohybu cívky v magnetickém poli. Magnetické pole je vytvářeno pomocí permanentních magnetů. Cívka je rozpořehobována zvukem a pohybem vytváří elektrický proud dle zákona elektromagnetické indukce. Dynamické mikrofony reagují na zvuk s výraznější nelinearitou než ostatní zmiňované mikrofony a jsou tedy vhodné na snímání zvuku ve specifickém frekvenčním rozsahu. [11] [12] [14]

2.1.3 MEMS mikrofon



Obrázek 1: Příklad MEMS mikrofonu

MEMS mikrofony mají membránu citlivou na zvuk vytvořenou přímo na křemíkovém waferu. Přímo s membránou je obvykle na čipu i předzesilovač. Digitální MEMS mikrofony mají v jednom čipu integrovaný spolu s mikrofonem a zesilovačem i analogově-digitální převodník. MEMS mikrofony obvykle fungují na stejném principu jako kondenzátorové mikrofony. [11] [14]

2.2 Metody digitalizace signálu

V předchozí kapitole byl popsán převod akustického zvukového signálu na analogový elektrický signál. Pro další zpracování počítačem je nutné signál digitalizovat, to znamená převést do nespojitě posloupnost digitálních údajů. Proces digitalizace se skládá ze dvou základních kroků, vzorkování a kvantování. Při digitalizaci dochází ke ztrátě informace. Množství ztracené informace se zmenšuje se zvyšující se frekvencí vzorkování a s rostoucím počtem bitů, do kterých je signál kvantován. [15] [16] [17] [18] [19] [24]

2.2.1 Vzorkování

Vzorkování je jedním ze dvou kroků digitalizace. Vzorkování znamená stanovování aktuální hodnoty signálu v pravidelných časových intervalech. Tímto procesem se signál rozdělí na vzorky. Každý vzorek obsahuje časový okamžik pořízení a analogovou hodnotu snímaného signálu v tomto okamžiku. Výstupem vzorkování je signál digitalizovaný v časové oblasti.[15] [16] [19] [24]

Vzorkovací frekvence by měla být zvolena úměrně požadované aplikaci. Většina obsahové informace v lidské řeči je přenášena na frekvencích nižších než 4 kHz. Dle Shannonova teorému je tedy vhodné vzorkovat vyšší vzorkovací frekvencí než 8 kHz. Frekvence vyšší než polovina vzorkovací frekvence je nutné odfiltrovat dolno propustným filtrem. Neodfiltrované frekvence vyšší než polovina vzorkovací frekvence jsou navzorkovány chybně. Tomuto jevu se říká aliasing. [6] [7] [23]

Pro záznam a přenos lidské řeči se nejčastěji používá vzorkovací frekvence 8, 16 a 32 kHz. Vzorkovací frekvence 8 kHz se používá pro vzorkování audia v telefonním hovoru. Audio navzorkované 8 kHz stačí pro přenos lidské řeči, potlačeny jsou však sykavky. Potlačení sykavek může

způsobovat problémy při analýze zvuku. Pro analýzu zvuku je vhodné využít vyšší vzorkovací frekvence, tedy 16, případně 32 kHz. [21]

2.2.2 Kvantování

Druhým krokem digitalizace je kvantování. Při kvantování je vzorek signálu, jehož hladina je hodnota ze spojitě množiny, přiřazen k nejbližší hladině z množiny nespojitých hodnot. Rozdíl mezi reálnou hodnotou a kvantizační hodnotou se nazývá kvantizační chyba. Kvantizační chyba se v digitálním signálu projeví jako šum. Audio technika obvykle využívá šestnácti bitové kvantování. Počet kvantovacích úrovní se označuje jako rozlišení převodníku. [22] [17] [18]

2.2.3 Analogově digitální převodník

Analogově digitální převodník je zařízení, které umí digitalizovat analogový signál. Existuje několik typů převodníků. Nejdůležitějšími parametry převodníků jsou vzorkovací frekvence a počet kvantovacích úrovní. Obecně platí, že se zvyšující se vzorkovací frekvencí klesá počet kvantovacích úrovní u převodníků stejné hodnoty. [24] [23] [25]

Tabulka 1: Porovnání rychlosti a rozlišení ADC

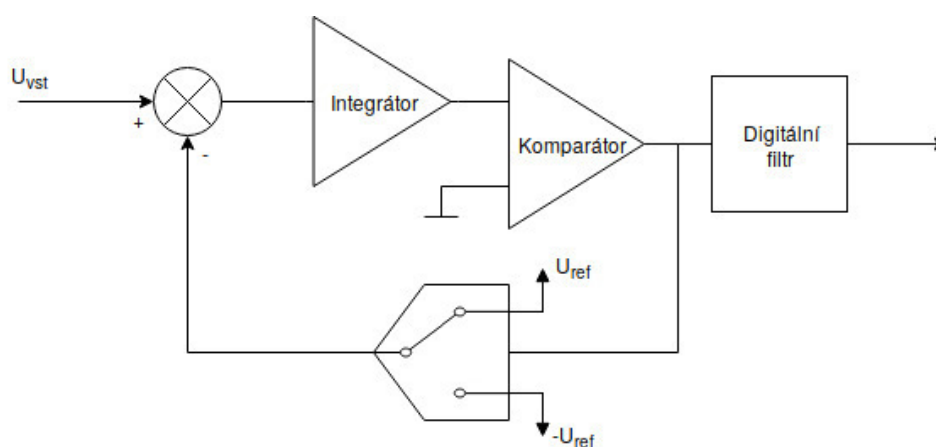
	Max rychlost	Max rozlišení [bit]
Paralelní (Flash)	10 GHz	8
S postupnou aproximací (SAR)	500 MHz	20
Sigma delta	600 kHz	24

Principiálně nejjednodušší z AD převodníků je paralelní převodník. Paralelní převodník potřebuje o jeden komparátor méně, než je jeho rozlišení. Na jeden vstup každého komparátoru je přiveden měřený signál. Na druhý vstup je přivedeno referenční napětí rovnoměrně rozdělené dle počtu komparátorů. Výstupy komparátorů jsou přivedeny do dekodéru, který signál z komparátoru převede na binární slovo. Převod pomocí paralelního převodníku probíhá v jednom kroku. Tyto převodníky jsou rychlé, ale s každým dalším bitem se zdvojnásobuje počet komparátorů v převodníku. Tyto převodníky obvykle nemají více než osm bitů a jejich vzorkovací frekvence je vysoce nad 16 kHz. Pro převod zvuku tedy nejsou vhodné. [23] [25]

Převodník s postupnou aproximací porovnává pomocí komparátoru měřené napětí s měnícím se referenčním napětím. Referenční napětí je nastavováno pomocí referenčního D/A převodníku. Na začátku cyklu je referenční napětí nastaveno na polovinu měřicího rozsahu. V případě, že měřené napětí je vyšší než referenční, je uložen nejvýznamnější výstupní bit převodníku na hodnotu jedna a nové referenční napětí je průměrem mezi měřícím rozsahem a původním referenčním napětím. V případě, že je měřené napětí v prvním kroku nižší než referenční napětí, je nejvýznamnější bit převodníku nastaven na hodnotu nula a nové referenční napětí je průměrem mezi původním referenčním napětím a nulou. Takto je v každém dalším kroku půlen měřicí rozsah a

hodnota na výstupu převodníku se přibližuje hodnotě na vstupu. Jelikož převod pomocí převodníku s postupnou aproximací není proveden v jednom kroku, je potřeba jeho vstup ošetřit proti změně napětí. K tomuto ošetření se používá obvod sample and hold. Převodníky s postupnou aproximací se dají pořídit s šestnácti bitovým rozlišením, s třiceti dvou už nejsou běžné. Jejich rychlost je vyšší, než je třeba u audio signálu. Jsou tedy použitelné, ale ne ideální. [23] [25] [26] [27]

Převodník typu sigma delta se skládá ze dvou bloků, modulátoru a digitálního filtru. Na vstupu modulátoru je rozdílový zesilovač. Na neinvertující vstup rozdílového zesilovače je přivedeno měřené napětí. Na začátku cyklu je na neinvertující vstup rozdílového zesilovače přivedeno referenční napětí jedné polaroty. Na výstupu rozdílového zesilovače je integrátor. Za integrátorem je komparátor, který porovnává výstup rozdílového zesilovače s potenciálem země. Výstupem komparátoru je logická jednička nebo nula. Tato hodnota je výstupem z modulátoru a zároveň je použita k překlopení polarity referenčního napětí. Překlopení polarity způsobí změnu směru nabíjení integrátoru. Poměr doby, kdy na výstupu modulátoru byla logická nula a doby, kdy na výstupu byla logická jedna, je úměrný napětí na vstupu převodníku vztaženému k referenčnímu napětí. Přepočet poměru doby logické jedničky a logické nuly na výstupu modulátoru na bitové slovo provádí digitální filtr. [28] [23] [25] [26] [27]



Obrázek 2: Schéma převodníku sigma delta

2.3 Metody rozpoznání řeči

Rozpoznání řeči je schopnost počítače převést lidskou řeč ve formě zvuku do formy, které rozumí počítač. Dle požadavku aplikace je vhodné zvolit správný typ rozpoznání řeči se správnými schopnostmi. Přehnané požadavky na funkčnost a schopnosti zvýší složitost a nároky na výpočetní výkon.

Systémy rozpoznání řeči mohou být závislé na řečnickovi, nezávislé na řečnickovi nebo adaptivní. Systémy závislé na řečnickovi jsou přizpůsobené jednomu řečnickovi. U ostatních fungují s menší přesností. Tyto systémy jsou obvykle jednodušší na vývoj a pro daného řečníka dosa-

hují lepších výsledků, než systémy nezávislé na řečnickovi. Systémy nezávislé na řečnickovi jsou schopny převádět řeč kohokoli. Tyto systémy jsou složitější na vývoj a jejich přesnost může být proměnlivá s řečníkem. Adaptivní systémy se mohou přizpůsobit hlasu řečníka po určitém učícím procesu. Učící proces může být například přečtení určitého textu. [30] [31] [32]

Systémy se také dělí dle vstupních dat. Systémy schopné rozpoznávat izolovaná slova umí převádět jen jednotlivá slova od sebe oddělená jiným signálem nebo dlouhou mezerou. Při rozpoznávání řeči je složité správně nalézt začátek slova. Tyto systémy jsou tedy jednodušší, ale nutnost oddělování slov je činí použitelnými jen v určitých aplikacích. [32]

Systémy schopné rozpoznávat slova v nepřetržitém projevu jsou složitější než systémy pro rozpoznání izolovaných slov právě z důvodu nutnosti správně separovat jednotlivá slova. Při převádění hlasu těmito způsoby může řečník mluvit téměř přirozeně. Systémy schopné převádět přirozenou řeč včetně mimoslovních projevů se nazývají systémy na rozpoznání spontánní řeči. [32] [33]

Detekce slova v projevu neboli „word spotting“ je dalším typem zpracování projevu. Detekce slova může být prováděna v nepřetržitém projevu i v izolovaných slovech. Systém nepřekládá projev, ale hledá v něm předem definovaná slova. Detekce slova je méně výpočetně náročná než plnohodnotné rozpoznávání řeči. Nevýhodou je omezená škála slov. Systém by se dalo přirovnat k slovníkovému systému s velmi malým slovníkem. [34]

Systémy se dále dělí dle velikosti slovníku. Systémy bez slovníku neřeší gramatiku a místo gramaticky správného zápisu slova používají svůj vlastní. V případě, že má systém porovnat detekované slovo s uloženým slovem, musí mít systém uložené slovo ve stejném tvaru, jako detekované slovo. Systémy se slovníkem hledají nejlepší fonetickou shodu detekovaného slova se slovy ve slovníku. Nejlepší shodu použijí jako výsledek. [32]

2.3.1 Metody detekce slova

V průběhu času byly pro detekci slov používány různé matematické mdely. Asi nejznámější z nich je skrytý Markovův model, ze kterého se na poli detekce slov začalo postupně ustupovat s rostoucím rozvojem umělé inteligence. Pro detekci slov byly nejprve používány konvoluční neuronové sítě, které se osvědčily na poli detekce obrazu. Zvukové okno bylo jednoduše převedeno na spektrogram a ten byl detekován sítí. Konvoluční neuronové sítě se však pro detekci slov neosvědčily tak dobře, jako na detekci obrazu. Poté přišly na řadu časově zpožděné neuronové sítě (TDNN), které opět zvýšily kvalitu detekce. V současné době se v disciplíně detekce slov ukázaly jako nejlepší rekursivní neuronové sítě, které byly vytvořeny speciálně na zpracování časově proměnných dat. [54]

Konvoluční neuronové sítě vykazují skvělých výsledků v oboru rozpoznání obrazu a technika jejich použití je dobře zvládnutá. Systémy na detekci slova skrze rozpoznání obrazu využívají konvoluční neuronovou síť tak, že zvukové úseky převádějí na obraz a ten používají jako vstupní data této sítě. Jako vstupní obraz neuronové sítě je obvykle použit spektrogram zvuku konstantní délky. K učení sítě je zapotřebí mít velké množství označených vzorků detekovaných slov. Tento

systém je výpočetně náročný z důvodu nutnosti převodu zvuku na obraz, ke kterému se obvykle používá Fourierova transformace. [46] [47] [48]

TDNN jsou konvoluční neuronové sítě navrženy tak, aby byly schopny zpracovávat časově proměnná data. Na vstup TDNN se data posílají po segmentech pokaždé, když je sejmut další vzorek. Znamená to, že síť zpracuje každý vzorek tolikrát, jaká je velikost jejího vstupního okna. Obvyklým výstupem sítě je foném, nicméně je možné síť sdružovat a výstupem potom může být slovo. TDNN se od klasické konvoluční neuronové sítě odlišuje tím, že sdílí hodnoty váhových vektorů ve vrstvě. Tím dojde k tomu, že když se samostatný vzorek pohybuje oknem, je sám o sobě rozpoznáván v každém kroku stejně. Jiná situace nastane, když vzorek není osamocený. V takovém případě je jeho rozpoznání závislé na něm samém a na okolních vzorcích, tedy kontextu širší informace. TDNN dosahují při rozpoznání hlasu lepších výsledků než klasické konvoluční neuronové sítě. TDNN jsou složitější na učení a výpočetně náročnější na provoz než klasické konvoluční neuronové sítě, existují však nástroje na zefektivnění výuky i běhu TDNN za cenu jen mírné ztráty kvality rozpoznávání slov. [49] [52] [51]

Rekurentní neuronová síť převádí výsledek z výstupu na svůj vstup v další iteraci. Znamená to tedy, že výsledek každé iterace rekurzivní neuronové sítě je kombinace vstupních dat a výsledku minulé iterace. Na rozdíl od tradičních konvolučních neuronových sítí mají rekurzivní sítě paměť, která jim umožňuje pracovat se vstupem s proměnnou délkou. Rekurzivní neuronové sítě najdou uplatnění tam, kde je potřeba generovat nebo analyzovat časově proměnná data jako je automatické generování novinových článků, automatické dokončování slov nebo rozpoznání řeči. Nevýhodou rekurzivní neuronové sítě oproti konvoluční je vyšší složitost a výpočetní náročnost učení sítě. Obecně však rekurzivní neuronové sítě dosahují lepších výsledků při rozpoznávání řeči než konvoluční neuronové sítě, včetně TDNN. [50] [54] [53]

2.4 Snowboy

Snowboy je nástroj na detekci klíčových slov v reálném čase od společnosti KITT.AI. Nástroj je poskytován zdarma pro nekomerční účely. Klíčové slovo je slovo nebo fráze, kterou se počítač snaží rozpoznat ve vstupním zvukovém signálu. Po rozpoznání klíčového slova počítač spustí určitou akci. Příkladem klíčového slova je „Alexa“ u Amazon Echo, „OK Google“ u Android aplikací nebo „Hey Siri“ u iPhone. U těchto produktů klíčová slova spustí výpočetně náročnější rozpoznávání všech slov. Klíčová slova však mohou být použita i v aplikacích, které reagují na omezenou množinu příkazů.[5]

Výhoda Snowboy oproti zařízení jako Alexa Smart Home je v tom, že si uživatel může vytvořit libovolné vlastní modely příkazů v libovolném jazyce. Další výhodou je, že Snowboy zpracovává zvuk lokálně. Zvuk není nikam poslán, takže nemůže dojít k narušení soukromí. Lokální zpracování je výhodné i proto, že k němu není potřeba připojení k internetu. Celý systém může být oddělený od internetu, což vylučuje případný útok z internetu.[5]

Snowboy nabízí dvě možnosti vytvoření modelu. První možnost je využít web „<https://snowboy.kitt.ai/>“. Po přihlášení může uživatel vyplnit formulář a nahrát třikrát příkaz. Po kontrole na-

hraní web vytvoří model, který si uživatel může stáhnout. Druhá možnost je HTTP dotaz POST na adresu „<https://snowboy.kitt.ai/api/v1/train/>“. Dotaz musí obsahovat jméno modelu, jazyk, ve kterém je model nahraný, věkovou skupinu a pohlaví autora, typ mikrofonu, token k ověření uživatele a třikrát nahraný příkaz ve wave formátu. Odpověď na dotaz POST je vytrénovaný model.[5]

2.4.1 Knihovna Snowboy

Knihovnu Snowboy je možné získat ve verzi pro operační systém Ubuntu, Andriod, iOS, OSX a Raspbian a pro Node.js. [5]

Před spuštěním detektoru je potřeba detektor inicializovat. Detektor potřebuje cestu ke konfiguračnímu souboru. Konfigurační soubor je neveřejný a tvůrci knihovny nezveřejnili jeho účel. Detektor potřebuje také textový řetězec cest k uživatelským modelům a citlivost knihovny na jednotlivé modely. Dále je možné nastavit zesílení mikrofonu pro případ, že má uživatel slabý mikrofon a povolit nebo zakázat automatické zesilování mikrofonu. Automatické zesilování mikrofonu je doporučeno používat pouze v případě univerzálního modelu s doporučením používat ho. V případě osobních modelů by povolení automatického zesilování mikrofonu mohlo zapříčinit zhoršení úspěšného rozpoznání klíčového slova. [5]

Kromě vedlejších funkcí, které vrátí aktuální nastavení detektoru provedené při inicializaci a formát, ve kterém detektor očekává data, obsahuje detektor dvě důležité veřejné funkce. Funkce „RunDetection“ má jako vstupní parametr část zvukových dat. Detektor při analýze data skládá za sebe a analyzuje. Funkce vrátí číslo -2 při nedetekování klíčového slova ani lidského hlasu. Při detekci lidského hlasu vrátí funkce číslo 0. Když při detekci dojde k chybě, je vrácena hodnota -1. V případě detekce klíčového slova je návratová hodnota index klíčového slova dle textového řetězce cest k uživatelským modelům. Klíčová slova v textovém řetězci jsou číslována od jedné. Detektor skládá za sebe části slov, které obdrží, když je volána funkce „RunDetection“. Takové chování může být nežádoucí v případě, kdy mu nejsou posílána všechna data. V případě posílání nekonzistentních dat je vhodné volat funkci „Reset“, která způsobí, že detektor nebere v úvahu data, která získal před voláním této funkce. [5]

2.5 Google asistent

Google asistent je multiplatformní hlasový asistent od firmy Google. Google asistent je následníkem Google now. Na rozdíl od Google now, který je schopen pouze plnit příkazy uživatele, Google asistent je schopen obousměrné konverzace. Společnost Google integruje Google asistent do svých chytrých zařízení. Další výhodou Google asistenta je, že byl vydán vývojový kit, který dovozuje třetím stranám implementovat Google asistent do svých zařízení nebo programů běžících na různých OS. Google asistent byl již implementován do Raspberry pi. Propojení Google asistent s vlastním zařízením je možné například přes servis IFTTT.

Google asistent na zařízení zvuk pouze nahrává a odesílá analyzačním serverům společnosti Google. Detekovaná slova jsou zaslána zpět do zařízení, které je analyzuje a dle nich vykoná úkon. Z důvodu, že analýza probíhá na analyzačních serverech Google, je potřeba aby zařízení bylo neustále připojeno k internetu. Kvůli tomu, že je vykonávána analýza překladu řeči na text, je potřeba specifikovat jazyk uživatele. Podporované jazyky jsou angličtina, francouzština, hindština, indonéština, němčina, italština, japonština, korejština, maráthština, portugalština, španělština, vietnamština, nizozemština, ruština, švédština, norština, dánština, tradiční čínština, tamilština, thajština, telugština, turečtina a polština. Každé zařízení, které využívá Google asistent musí být zaregistrované u společnosti Google. Registrace zařízení probíhá online a pro nekomerční užití je zdarma. Pro komerční účely je třeba souhlasu společnosti Google. [35] [36]

2.6 Alexa voice service

Alexa voice service je aplikační rozhraní firmy Amazon, které umožňuje výrobcům integrovat možnosti Alexa do jejich zařízení. Alexa je hlasový asistent s portfoliem schopností srovnatelnými s Google asistent. Amazon dovoluje vývojářům třetích stran programovat vlastní funkce hlasového asistenta. Tyto funkce jsou nazvány „skills“. Funkce může být například určitá reakce na určitou frázi nebo hra jako vědomostní kvíz. Příkazy, vlastní funkce a vlastní zařízení je možné propojit přes služby jako IFTTT nebo přes službu ACK. Podporované jazyky asistenta Alexa jsou angličtina, němčina, francouzština, italština, španělština a japonština. [37] [38] [39] [40] [41] [42]

Zpracování řeči probíhá na serverech Amazonu. Zařízení, které chce využívat hlasového asistenta Alexa, musí být připojeno k internetu a musí obsahovat identifikační token. Užití Alexa voice service je zdarma, využití pro komerční účely je podmíněno souhlasem společnosti Amazon. [41] [42]

2.7 Sopare

Sopare je open source projekt, který si klade za cíl vytvořit nástroj k offline detekci klíčových slov na jednodeskových počítačích. Pomocí nástrojů Sopare je možné vytvořit model, vytrénovat ho a poté jej detekovat v hlasovém projevu. Sopare nedělá překlad zvuku na text, tudíž ho je možné použít v jakémkoli jazyce. Díky velkému portfoliu nastavení zvládá i detekci jiných zvuků, než jsou slova. Například může detekovat ťukání, tleskání. Detektor má řadu nastavení, díky kterým se dá zvýšit správnost a přesnost detekce. Nastavení „Margin value“ určuje, jak velká část slova se musí shodovat s modelem. Nastavení „Cross similarity“ určuje, vyžadovanou míru podobnosti mezi trénovaným modelem a zpracovávaným zvukem. Nastavení „Left distance“ a „Right distance“ určuje, jak podobný je model se zpracovávaným zvukem ve frekvenční oblasti. „Left distance“ se zaměřuje na nízké frekvence, „Right distance“ na vysoké. Nastavení „Cross similarity“ se navíc dělí na tři nastavení. Nastavení „Norm“ je porovnání FFT, nastavení „Height“ je porovnání hlasitosti a nastavení „Domain frequency“ slouží k nastavení porovnání základ-

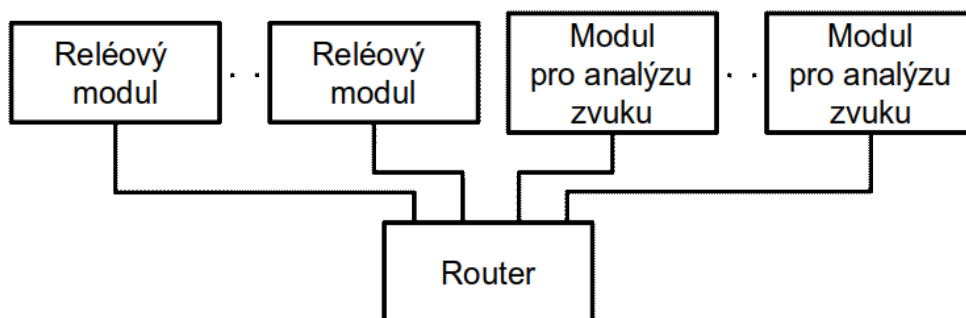
ních frekvencí modelu a zpracovávaného zvuku. Vytvořený model je možné trénovat opakovaně, s každým tréninkem se zvyšuje přesnost a správnost detekce modelu. Trénink i detekce probíhá offline. K detekci je doporučeno použít výkonnější jednodeskové počítače jako je Raspberry pi 3. [43] [44]

3 Návrh vestavěného řídicího systému pro analýzu základních slov řečového signálu

Cílem práce je navrhnout, realizovat a verifikovat systém pro detekci slov, použitelný například v domovní automatizaci. Systém má analyzovat zvuk z okolí a v případě, že detekuje klíčové slovo, má na něj zareagovat změnou stavu výstupu. Uživatel má být schopen nastavit propojení klíčového slova s výstupem, přidávat klíčová slova a navyšovat počet výstupů.

Je potřeba, aby zařízení bylo umístěno na doslech příkazů a zároveň bylo schopno spínat zařízení, která mohou být různě rozmístěna nebo světla, která mívají přivedené elektrické vývody do rozvaděče. Kvůli této podmínce by se případné použití samostatného zařízení značně zkomplikovalo připojování zařízení k řídicímu systému, a proto byl systém navrhnut jako dvě samostatná zařízení propojena LAN sítí.

Dalším prvkem systému je uživatelský program, který slouží k nastavení a propojení těchto zařízení mezi sebou a s klíčovými slovy. Obrázek 3 popisuje navrhovanou konfiguraci systému.



Obrázek 3: Schéma propojení částí systému

Zařízení, s nimž uživatel přijde nejvíce do styku, je modul pro detekci klíčového slova v řečovém projevu, dále označovaný jako „Hlasový ovladač“. Hlasový ovladač je umístěn v obývané místnosti na doslech příkazů. Předpokládané vhodné umístění Hlasového ovladače může být například noční stolek, vršek lednice, okenní římsa apod. Hlasový ovladač má snímat zvuk z okolí, analyzovat ho a v případě rozpoznání příkazu na tento příkaz reagovat vysláním zprávy reléovému modulu.

Další zařízení je reléový modul ovladatelný přes místní síť, dále nazvaný „Spínač“. Spínač má sloužit ke spínání zařízení, která jsou k němu připojena. Spínače mají být ovládány Hlasovými ovladači. Každý Spínač má být schopen spínat pět až deset zařízení.

Počet Hlasových ovladačů a Spínačů má být dostatečný pro řízení osvětlení rodinného domu.

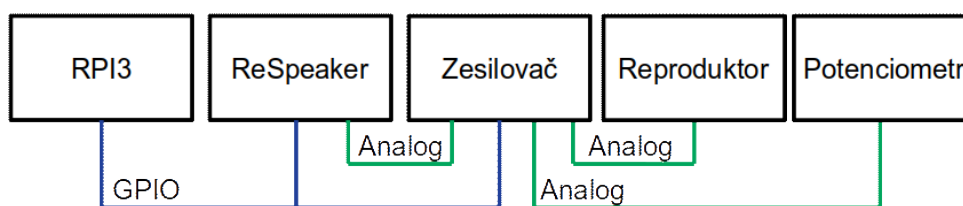
Součástí systému má být také uživatelský program pro konfiguraci Hlasového ovladače a Spínačů, dále „Uživatelský program“. Pomocí Uživatelského programu má být možné vytvářet nové hlasové modely, tyto modely propojovat s výstupy konkrétních Spínačů a takto vytvořenou konfiguraci nahrát do Spínačů a Hlasového ovladače.

3.1 Návrh hlasového ovladače

Jako výpočetní jednotka Hlasového ovladače byl zvolen jednodeskový počítač Raspberry pi 3. RPI3 má dostatečný výpočetní výkon pro tento typ aplikací a disponuje rozhraním Ethernet a wi-fi, která mohou být použita pro propojení se Spínači.

Pro snímání a digitalizaci zvuku byl zvolen modul ReSpeaker 2-Mics Pi HAT. Modul byl zvolen, neboť je kompatibilní s RPI3, obsahuje audio kodek a dva mikrofony. Modul navíc obsahuje diody, které mohou být použity jako vizuální zpětná vazba uživateli. Kvůli akustické zpětné vazbě má být vyvinut modul s audio zesilovačem, ke kterému se připojí reproduktor a potenciometr na nastavení hlasitosti.

Všechny desky jsou propojeny sběrnici GPIO. Zesilovač a ReSpeaker navíc ještě audio konektorem 3,5 mm Jack. Celkové schéma propojení modulů a RPI3 je na obrázku 4.



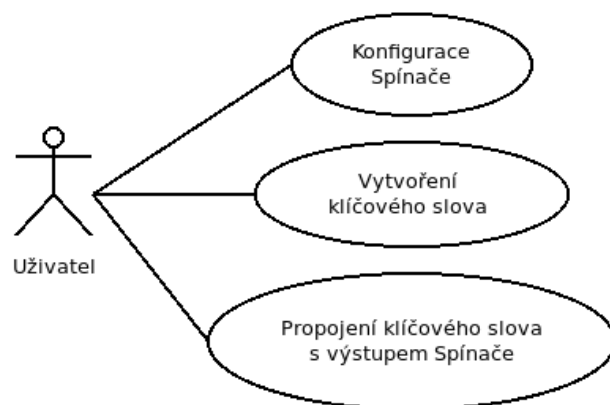
Obrázek 4: Návrh propojení modulů

3.2 Návrh Spínače

Spínač je modul, který slouží ke spínání zařízení k němu připojených. Spínač má dostávat příkazy k sepnutí výstupu od Hlasového ovladače přes LAN síť. Výstupem Spínače mají být piny COM a NO pole relé. Spínač tak bude možné použít přímo ke spínání připojeného zařízení nebo ke spínání signálového vodiče. Vhodné umístění Spínače je v rozvodné skříni. K tomuto umístění byla navržena i krabice Spínače, která má uchycení na DIN lištu. Spínač tedy musí obsahovat výpočetní prvek, ethernetový řadič, řadič USB, reléové pole a napájecí obvod.

3.3 Návrh uživatelského programu

Uživatelský program slouží ke konfiguraci Spínačů a Hlasového ovladače. Programem má jít konfigurovat způsob připojení modulů do místní sítě. Pomocí programu má být také možné vytvářet nové modely a tyto modely přiřazovat konkrétním výstupům Spínače. Funkce jsou zobrazeny diagramem použití na obrázku 5.



Obrázek 5: Návrh užití uživatelského programu

4 Realizace vestavěného řídicího systému pro analýzu základních slov řečového signálu

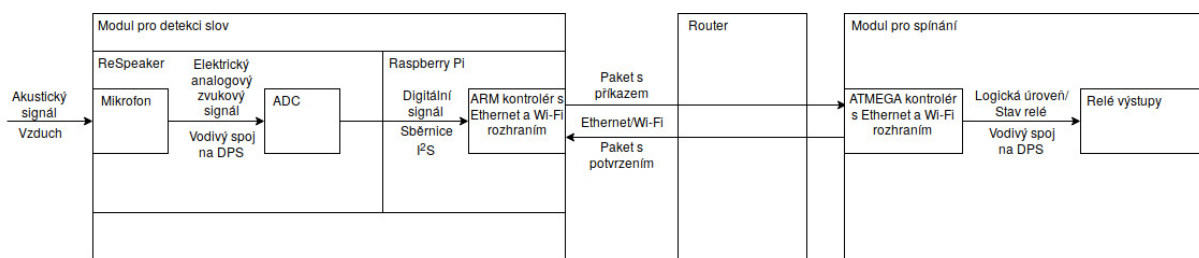
Pro realizaci Hlasového ovladače byla zvolena vývojová platforma RP3 díky jejím kompaktním rozměrům a předpokládanému dostatečnému výkonu. Pro snímání zvuku byla zvolena platforma ReSpeaker. ReSpeaker se osvědčil jako dostatečný v podobných aplikacích a žádný z jeho parametrů nenaznačoval, že by mohl být pro tuto aplikaci nevhodný. Zesilovací obvod byl zvolen tak, aby dokázal s přiměřenou kvalitou ozvučit běžnou místnost. Systém není určen k přehrávání hudby, pouze k akustickému upozornění uživatele.

K realizaci softwerové stránky byl zvolen operační systém Raspbian, který je doporučen výrobcem RP3. Pro tento systém je zkompileována nebo je možné zkompileovat knihovnu PortAudio, která slouží ke snímání zvuku a je dostačující pro tuto aplikaci. Pro systém je zkompileována také knihovna Snowboy, která detekuje klíčová slova ve snímaném zvuku. K programování byl zvolen jazyk C++.

K realizaci Modulu pro spínání byl zvolen mikrokontrolér Atmega328 z důvodu jeho rozšířenosti a jednoduchosti vývojové platformy Arduino. Atmega328 obsahuje dostatečné množství výstupů a disponuje dostatečným výkonem pro tuto aplikaci.

Jako komunikační prostředek pro konfiguraci zařízení byly zvoleny běžně rozšířené technologie USB a Ethernet. Ke komunikaci mezi zařízeními byla zvolena místní síť. Toto řešení by mělo snížit na minimum složitost instalace zařízení, protože většina uživatelů již touto sítí disponuje.

Tok informace do systému a v systému popisuje diagram na obrázku 6.



Obrázek 6: Transformace informace systémem

4.1 Hardware Raspberry Pi

Raspberry Pi 3 Model B+ je hlavní periferií Hlasového ovladače. O bezproblémový chod periferií hlasového ovladače se stará operační systém Raspbian 8.0 (jessie). Raspbian je operační systém postavený na systému Debian a je optimalizovaný pro jednodeskové počítače RPI3. Všechny periferie hlasového ovladače jsou napájeny skrz RPI3. Napájecí port je typ USB 2.0 Micro B. Raspberry disponuje konektorem RJ45, který je možné využít pro propojení Hlasového ovladače a Spínačů přes LAN síť. Ke stejnému účelu slouží i Wi-Fi modul.

Propojení RPI3, modulu ReSpeaker a modulu zesilovače zajišťuje čtyřiceti pinový GPIO port. Port obsahuje Dva piny s napájecím napětím, dva piny s napětím 3,3 V, osm zemnicích, dva rezervované piny a dva piny pro sériové komunikační rozhraní UART. Ostatní piny jsou 3,3 voltové digitální vstupně výstupní piny s maximálním proudovým odběrem 16 mA/pin a celkovým maximálním odběrem 30 mA. Dva digitální piny na sběrnici GPIO jsou zároveň piny pro komunikační sběrnici I2C. Pět digitálních pinů může být použito jako sběrnice SPI0 a Čtyři další jako směrnice SPI1.

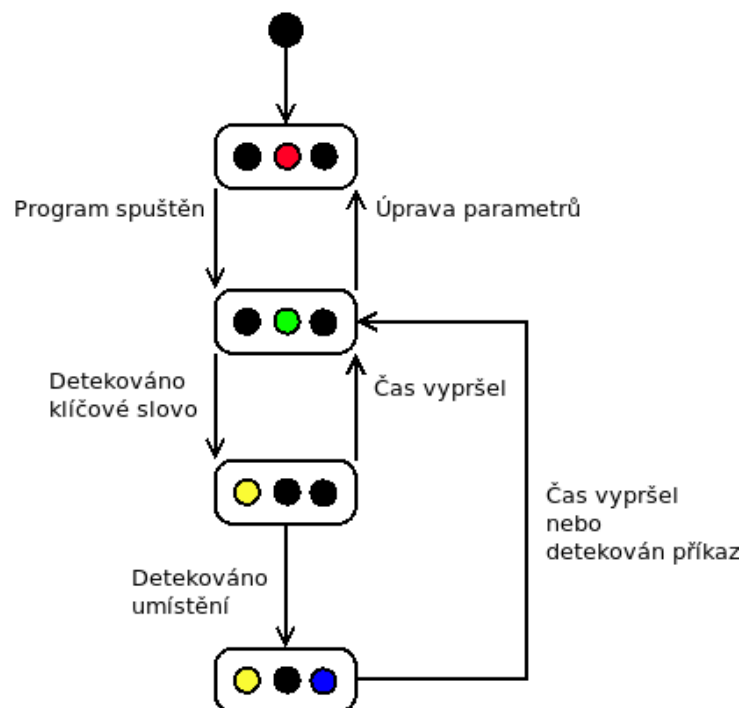
Raspberry Pi 3 obsahuje čtyř jádrový procesor Broadcom BCM2837 ARM, pracující na frekvenci 1,2 Ghz a 1 Gb operační paměti. Součástí Raspberry je také integrovaná grafická karta Broadcom VideoCore IV. [29]

Doporučené minimální napájení pro Raspberry Pi 3 je 2,5 A, 5,1 V stejnosměrných. Napájecí napětí nesmí stoupnout nad 5,25 V ani klesnout pod 4,75 V. [1]

4.2 Hardware ReSpeaker

ReSpeaker 2 Mic Array je modul pro RPI3 vytvořený pro aplikace rozpoznání řeči. Modul je postaven na nízko odběrovém stereo kodeku WM9860. Zařízení disponuje dvěma mikrofony, třemi RGB LED, uživatelským tlačítkem a 3,5 mm Audio Jackem.

Uživatelské RGB LED jsou tři zařízení s označením apa102. Barva všech LED je nastavována přes sběrnici SPI, ke které jsou LED připojeny. SPI je součástí GPIO. LED obsahují vlastní řídicí obvod, který udržuje nastavenou barvu po dobu napájení. Díky tomu není zatěžován procesor RPI3 a také není nutné vytvářet speciální vlákno na obsluhu LED. Změna barev jednotlivých LED je popsána diagramem na obrázku 7.



Obrázek 7: Změny barev tří RGB LED při běhu programu

Uživatelské tlačítko umístěné na desce ReSpeaker slouží k opětovnému vyhledání 4.5.1 a připojení všech Spínačů ve stejné LAN síti. Tlačítko je připojené přes pull-up rezistor na GPIO sběrnici, pin GPIO17.

ReSpeaker disponuje dvěma mikrofony MSM421A3729H8-C, které jsou připojeny k audio kodeku WM8960. Komunikace mezi Audio kodekem a RPI3 je zajištěna pomocí sběrnice I²S, která je součástí GPIO. Audio kodek podporuje všechny vzorkovací frekvence mezi 8 kHz a 48 kHz.

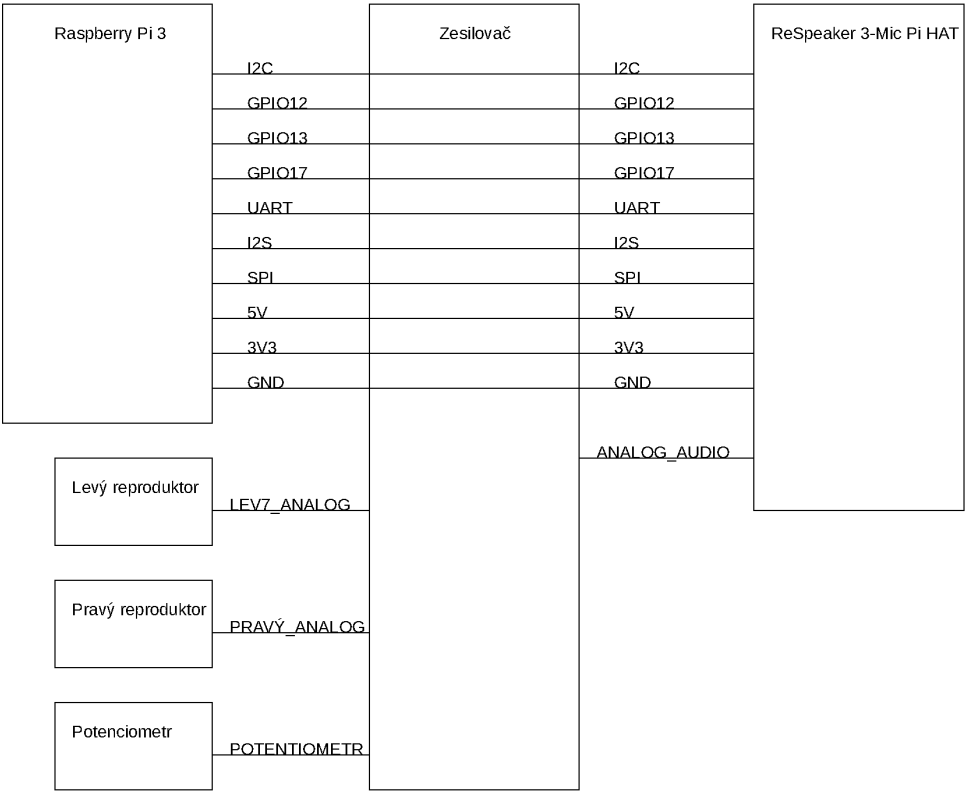
4.3 Hardware Zesilovače

K Zesilovači je připojen modul ReSpeaker pomocí konektoru 3,5 mm jack. Zesilovač zesiluje výstupní zvukový signál z Audio kodeku do dvou 2 W reproduktorů. Hlavní komponentou desky Zesilovač je audio zesilovač PAMA8403. Spolu s reproduktory je k zesilovači připojen i potenciometr na ovládání hlasitosti. Reprodukory i potenciometr jsou vyvedeny mimo desku pro dosažení vyššího uživatelského komfortu. Propojení všech komponent vychází z doporučeného zapojení zesilovače PAMA8403.

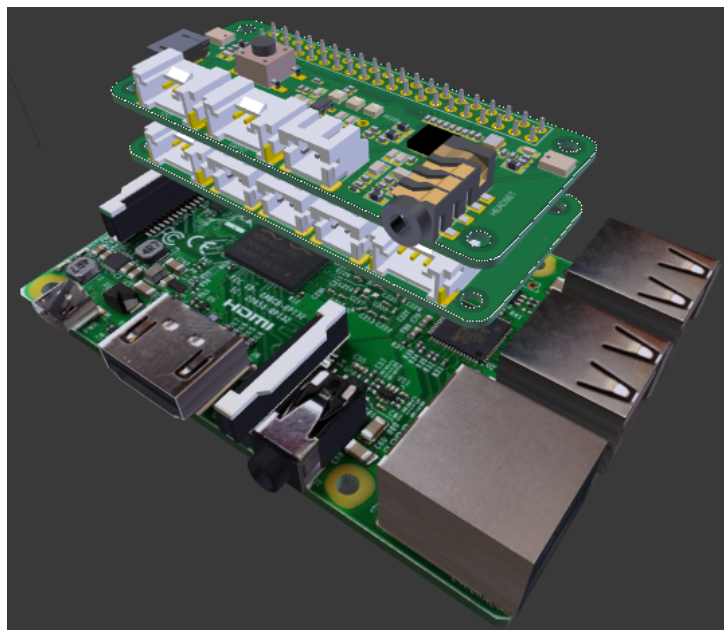
Stejně jako ReSpeaker je zesilovač připojen na GPIO. Pomocí GPIO je deska napájena. Schéma Zesilovače a DPS vytvořená v programu Eagle je součástí přílohy A, Zesilovač.

4.4 Propojení RPI3, ReSpeakeru a Zesilovače

Propojení modulů znázorňuje obrázek 8. Všechny moduly jsou propojeny sběrnici GPIO. ReSpeaker využívá komunikačních i napájecích vodičů sběrnice, Zesilovač pouze napájecích. Zesilovač a ReSpeaker jsou navíc propojeny audio kabelem. Na obrázku 9 je znázorněno uspořádání modulů.



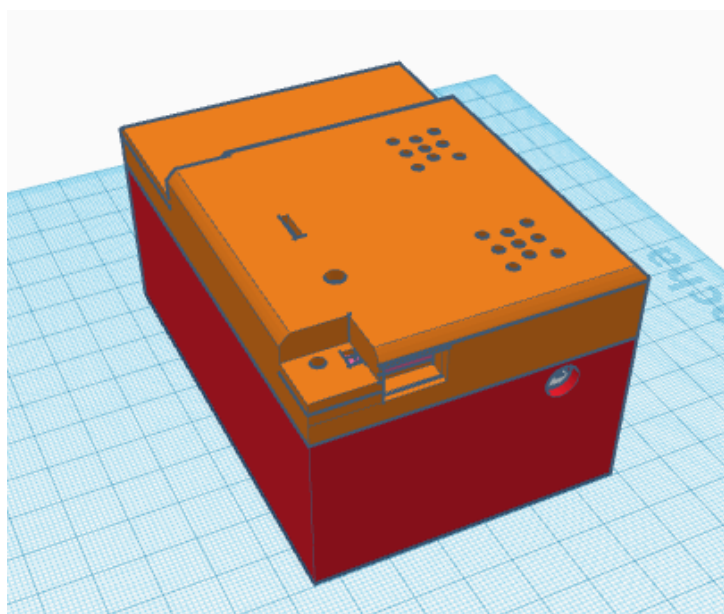
Obrázek 8: Schéma propojení modulů a Raspberry pi



Obrázek 9: Propojení zesilovače, ReSpeaker a Raspberry pi

4.4.1 Krabíčka

Design krabíčky zohledňuje její předpokládané umístění na viditelném místě, nutnost připojení napájení, možnost připojení ethernetového kabelu, potřebu díry na potenciometr, potřebu děr na reproduktory, LED a uživatelské tlačítko a také nutnost děr na mikrofony. Model krabíčky je na obrázku 10 a v příloze A, Model Krabíčky.



Obrázek 10: Model krabíčky Hlasového ovladače

4.5 Software Hlasového ovladače

Program Hlasového ovladače běží v operačním systému Raspbian na jednodeskovém počítači Raspberry pi. Program je spuštěn po naběhnutí operačního systému službou systemd a sám o sobě nikdy neskončí. Nezkompilovaný program je v příloze A, Program na detekci slova - projekt. Program zkompilovaný pro Raspbian je v příloze A, Program na detekci slova - aplikace.

Hlavní funkcí programu Hlasového ovladače je reakce na klíčová slova. O nahrávání zvuku se stará třída PortAudio. O detekci klíčových slov se stará třída Snowboy. Fungování programu rámcově popisuje diagram na obrázku 11.

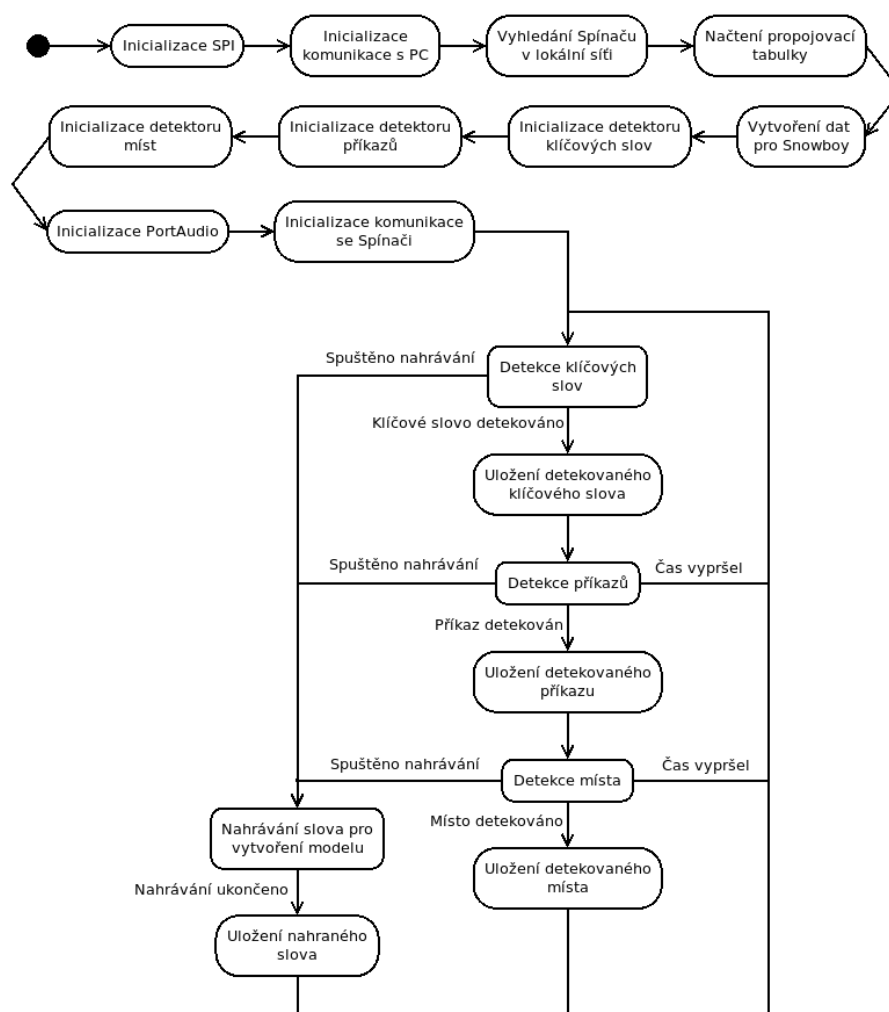
4.5.1 Startovací sekvence

V případě, že je program spuštěn s parametrem, vypíše, k čemu slouží, jak ho spustit, autora a datum vytvoření. Následuje inicializace SPI, které obstarává knihovna wiringPi. Poté je volána funkce getNetworkStats, která ověří připojení k místní síti. Následně je spuštěno vlákno myPc-CommunicationThread, viz 4.5.2. Následuje vyhledání Spínačů funkcí ArduinoFinder a poté načtení propojovací tabulky. Data z propojovací tabulky jsou vložena do tří vektorů. Vektory obsahují strukturu Hotwords, která se skládá z názvu klíčového slova a citlivosti. Trojice Stringů vytvořených z těchto vektorů je vstupem třídy SnowboyDetect. Následuje vytvoření vektoru propojů místa s digitálním výstupem konkrétního spínače. K tomuto účelu slouží vektor propojů datového typu Connections. Connections obsahuje index klíčového slova, které spojení vyvolá, jméno Spínače, digitální výstup spínače, režim výstupu a délku pulsu.

Následuje vytvoření instancí třídy SnowboyDetect nazvané jako detector1, detector2 a detector3. Seznam modelů pro detector1 obsahuje klíčová slova neboli oslovení programu. Účelem klíčového slova je snížení šance na falešný alarm. Seznam modelů pro detector2 obsahuje slova příkazu, například „Zhasni“ nebo „Zapni“. Detektor3 obsahuje modely umístění ovládaného zařízení nebo zařízení samotného, například „kuchyň“, „topení“.

V dalším kroku je inicializována třída PortAudioWrapper, která zajišťuje nahrávání audia. Dále je vytvořeno vlákno communicationThread 4.5.4 a program vstupuje do smyčky, ve které je periodicky volána detekce klíčového slova. Po detekci klíčového slova je tato detekce zastavena a je spuštěna detekce příkazu. Po detekování příkazu je zastavena detekce příkazu a spuštěna detekce místa. V případě, že po dobu šesti sekund od detekce klíčového slova není detekován příkaz, program se vrací k detekci klíčového slova. V případě, že šest sekund po detekci příkazu není detekováno místo, vrací se program k detekování klíčového slova. Kteroukoli detekci může přerušit příkaz na nahrávání slova z PC programu 3.3.

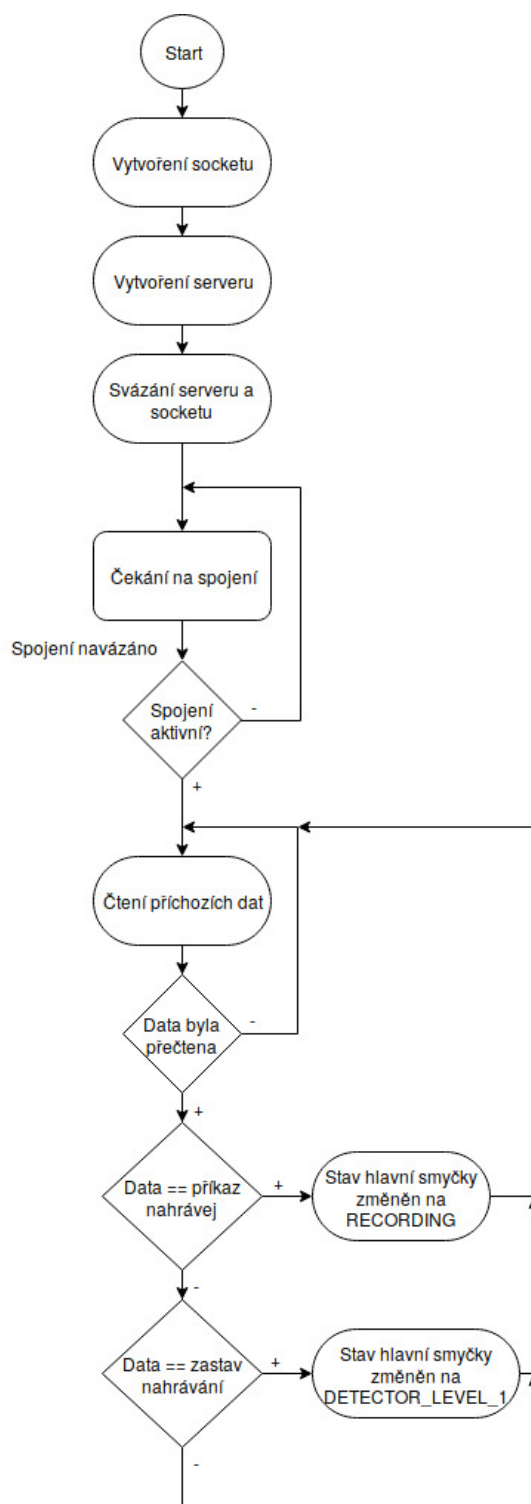
V režimu nahrávání slova se vytvoří soubor s příponou wav, do kterého jsou vkládána výstupní data ze třídy PortAudioWrapper. Nahrávání je zastaveno příkazem z PC programu nebo po deseti sekundách od začátku nahrávání. Po ukončení nahrávání slova se program vrací k detekci klíčového slova.



Obrázek 11: Diagram startovací sekvence programu a rozpoznávání klíčových slov

4.5.2 Komunikace s PC

Komunikaci s PC zprostředkovává vlákno `myPcCommunicationThread`, které funguje jako TCP/IP server na portu 8888. Vlákno je spuštěno krátce po startu programu. Fungování vlákna popisuje diagram na obrázku 12.



Obrázek 12: Diagram popisující vlákno `myPcCommunicationThread`

4.5.3 Vyhledání Spínačů

Funkce `ArduinoFinder` slouží k vyhledání Spínačů v místní síti. Funkce je volána krátce po spuštění Hlasového ovladače a po stisknutí uživatelského tlačítka. Funkce nejprve zjistí svou IP adresu a tím i informaci, ve které lokální síti se nachází. V téhle síti se funkce pokusí postupně připojit k serveru na každé adrese. V případě, že se funkci podaří navázat spojení, odešle serveru zprávu s dotazem na typ a jméno zařízení. Spínače na tento dotaz odpoví a jejich jméno a IP adresa je uložena do seznamu IP adres. V případě, že se funkci nepodaří připojit se nebo přijmout odpověď v odpovídajícím tvaru, postupuje k pokusu o připojení k serveru na další adrese v řadě. Návrátová hodnota funkce udává počet nalezených Spínačů v lokální síti.

4.5.4 Reakce na detekovaná slova

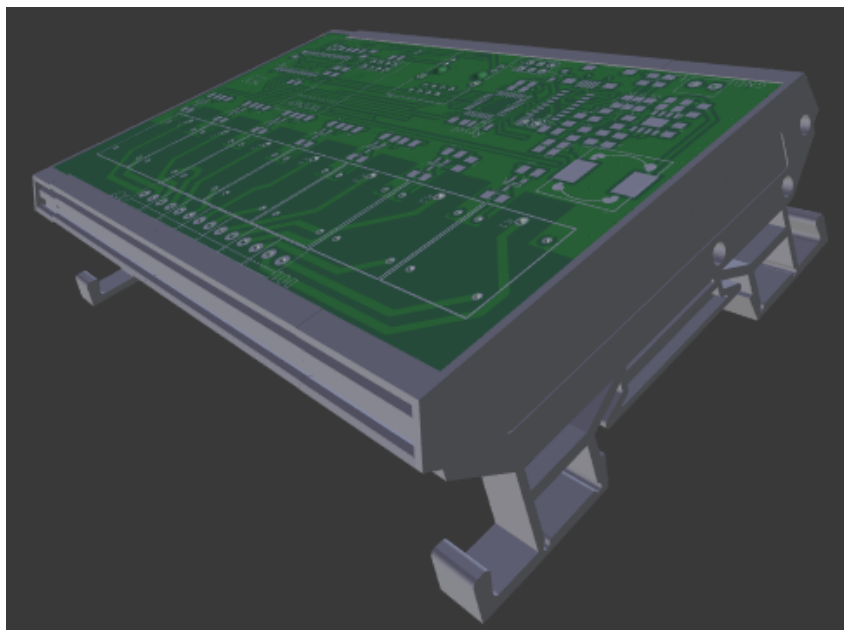
Vlákno `communicationThread` se stará o reakci na detekovaná slova detektory a o změnu barev LED dle diagramu na obrázku 7. Po spuštění třída skočí do nekonečné smyčky. Ve smyčce se ověřuje, zda bylo zmáčknuto uživatelské tlačítko. Pokud bylo, jsou vyhledány Spínače v lokální síti. Po obsluze tlačítka je z bufferu odebráno nejstarší detekované slovo. V případě, že detekované slovo je příkaz, vyhledá se tento příkaz v propojovací tabulce a uloží se informace, jestli se má poslat vypínací nebo zapínací příkaz do Spínače. V případě, že je detekované slovo umístění spínaného zařízení, je volána funkce `MessageSend`, viz 4.5.5. Poté je vlákno uspáno na 200 ms.

4.5.5 Komunikace se Spínači

`MessageSend` slouží k poslání příkazu Spínači. Vstupními parametry funkce je informace, jestli se má provést Sepnutí nebo rozepnutí Spínače a číslo spojení. Spojení jsou vytvářena po spuštění programu z propojovací tabulky a obsahují propojení slov, umístění spínaného zařízení, jméno Spínače, který zařízení spíná, informaci, jestli se má provést sepnutí nebo puls a délku pulsu. Funkce `MessageSend` vytvoří pomocí konkrétního Spojení zprávu pro Spínač. Poté je vytvořen socket, vyhledá se IP adresa Spínače v seznamu IP adres. Naváže se spojení se Spínačem a odešle se vytvořená zpráva.

4.6 Hardware Spínače

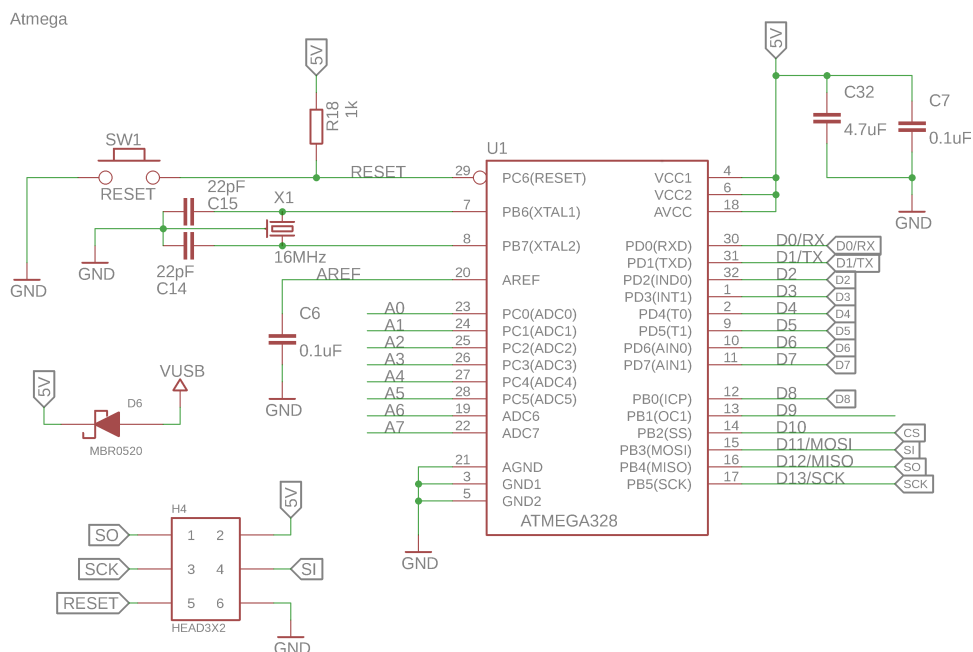
Hardware spínače se dá rozdělit do pěti základních částí. Výpočty zajišťuje obvod s mikrokontrolérem `Atmega328`. K převodu komunikace RS232 na USB slouží obvod s převodníkem `CH340G`. Komunikaci po síti LAN zprostředkovává Ethernetový řadič `ENC28J60`. Napájení 5 volty má na starost DC/DC konvertor s čipem `MP2307DN`. Výstupní uživatelská relé jsou spínána mikrokontrolérem přes unipolární tranzistory. Schéma Spínače a DPS vytvořená v programu `Eagle` je součástí přílohy A, Reléový modul. Spínač je umístěn v krabici na DIN lištu. DPS v krabici je na obrázku 13.



Obrázek 13: Model DPS Spínače s krabičkou na DIN lištu

4.6.1 Výpočetní část Spínače

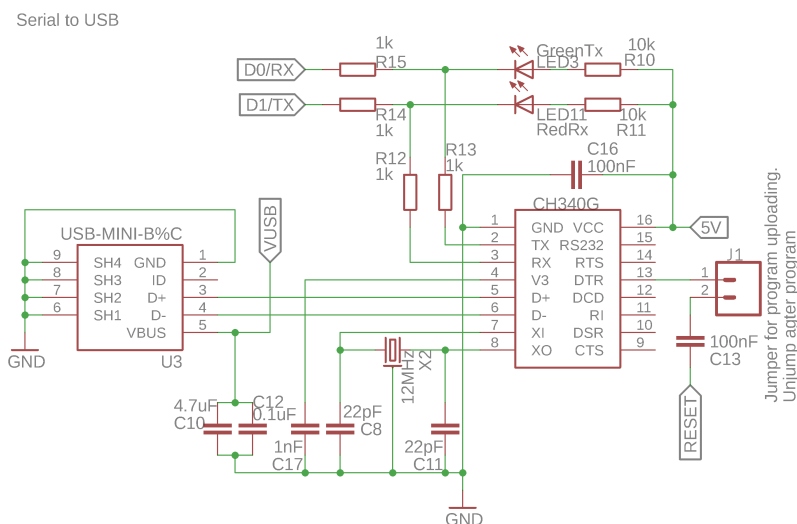
Výpočetní operace zajišťuje mikrokontrolér Atmega328. Při návrhu bylo vycházeno z doporučeného zapojení. Schéma zapojení je na obrázku 14.



Obrázek 14: Schéma zapojení mikrokontroléru Atmega328 ve Spínači

4.6.2 Převodník USB na sériovou linku

Součástí Spínače je mikro USB konektor, který slouží ke konfiguraci výrobku. Mikrokontrolér Atmega328 nedisponuje USB ovladačem, pouze sériovou linkou. Jako interface mezi mikrokontrolérem a počítačem slouží převodník CH340G. Schéma zapojení vychází ze zapojení doporučeného výrobcem čipu.



Obrázek 15: Schéma zapojení převodníku CH340G ve Spínači

4.6.3 Ethernetové rozhraní

Připojení mikrokontroléru k LAN síti zabezpečuje Ethernetové rozhraní ENC28J60. Ethernetové rozhraní je zapojeno dle doporučení výrobce.

4.6.4 Napájení 5 V

The diagram shows a step-down converter circuit. The input is labeled '6-20V to 5V/2A'. The output is labeled '5V'. The circuit includes an input filter with capacitors C31 and C1, a feedback network with resistor R4 and capacitor C4, a compensation network with capacitor C3 and resistor R3, an output filter with inductor L1 and capacitors C2 and C30, and a load represented by a blue LED (LED1). The IC is labeled 'MP2307DN-LE-Z'.

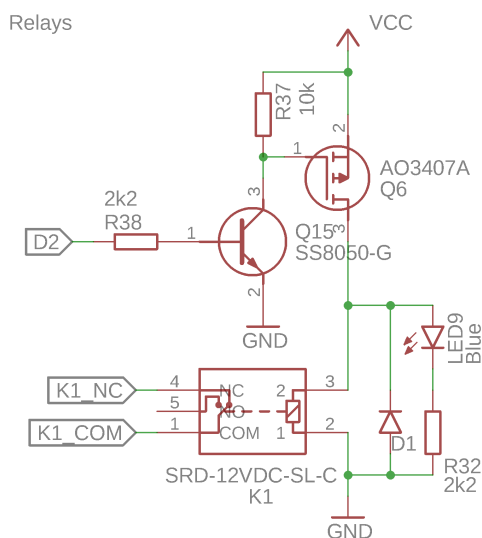
39

4.6.5 Napájení 3,3 V

Napájení 3,3 V zabezpečuje napěťový regulátor LM1117MPX. Regulátor napájí čip CH340G, který má maximální spotřebu 180 mA. Napětí je přivedeno na jednu z komunikačních linek přes pull up resistor omezující odběr této linky na 66 mA. Celkový maximální proud, který regulátor musí dodat, je 246 mA. Při napěťovém rozdílu 1,7 V je ztrátový výkon na tomto regulátoru 0,425 W. Dle datasheetu regulátoru tento proud nepřekračuje maximální odběr regulátoru a teplo vzniklé ztrátovým výkonem 0,425 W je regulátor schopen odvést. K regulátoru jsou připojeny pouze filtrační kondenzátory dle doporučení výrobce.

4.6.6 Výstupní relé

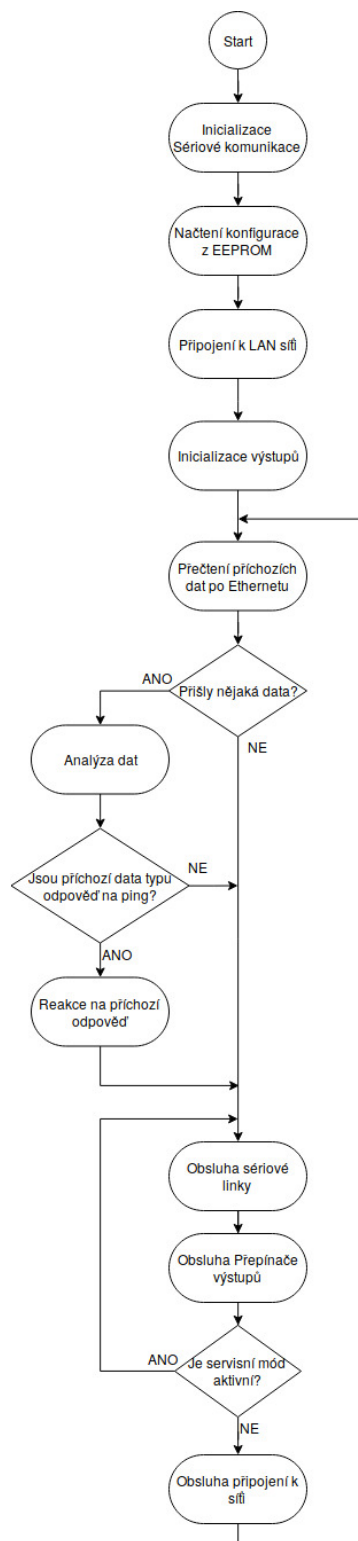
Spínač obsahuje sedm výstupních relé. Relé jsou napájena napájecím napětím. Relé jsou spínané tranzistory typu P-MOSFET. Tranzistory typu P-MOSFET jsou spínány mikrokontrolérem přes tranzistory NPN. Paralelně k relé jsou připojeny indikační diody a ochranné diody v závěrném směru. Schéma jednoho ze sedmi výstupních relé je na obrázku 18.



Obrázek 18: Schéma zapojení výstupního relé

4.7 Software Spínače

Životní cyklus programu popisuje blokově diagram na obrázku 19. Program Spínače je napsán v aplikačním rámci Arduino. Arduino je aplikační rámec pro vývoj aplikací v jazyce C++ pro kontrolery ATmega. Program Spínače je v příloze A, Program Spínače - projekt.



Obrázek 19: Blokové schéma životního cyklu programu Spínač

4.7.1 Inicializace Sériové komunikace

Inicializaci i všechny ostatní činnosti týkající se sériové komunikace zajišťuje nativní třída Serial. V kroku Inicializace je tedy pomocí této třídy spuštěna seriová komunikace s rychlostí 57600 Bd.

4.7.2 Načtení konfigurace z EEPROM

Načtení konfigurace z EEPROM je prováděno pomocí nativní třídy EEPROM. Tato třída umožňuje ukládat data do paměti EEPROM kontroleru Atmega a tato data z ní vyčítat. V kroku Načtení konfigurace z EEPROM je přečtena část paměti, která byla určena pro uchování konfigurace. Z paměti je vyčteno jméno Spínače; informace, jestli má být aktivní DHCP; IP adresa Spínače a IP adresa implicitní brány. Formát uložených dat v EEPROM znázorňuje tabulka 2.

Tabulka 2: Data uložená v paměti EEPROM

Byte	0	1	2	3	4	5	6	7	8	9	10	11	12
Význam	Jméno Spínače				Statická IP adresa				DHCP	IP adresa brány			

4.7.3 Připojení k LAN síti

O spojení s LAN sítí se stará knihovna EtherCard. Při připojování do sítě je nejprve potřeba se spojit s ethernetovým rozhraním. Program se o spojení pokouší periodicky a po každém neúspěšném pokusu o spojení obsluhuje sériovou linku. Po spojení s ethernetovým rozhraním program dle vyčtené konfigurace zažádá o přidělení IP adresy DHCP serverem a následně se pokusí připojit do sítě, nebo se pokusí připojit do sítě se statickou IP adresou. Připojení a získání IP adresy z DHCP serveru zajišťuje knihovna EtherCard.

4.7.4 Inicializace výstupů

Vstupně výstupní digitální porty 2 až 9 jsou inicializovány funkcí pinMode jako digitální výstupy. Tyto výstupy ovládají výstupní relé.

4.7.5 Přečtení výstupních dat po Ethernetu

Knihovna EtherCard dokáže na základě požadavku vrátit velikost příchozích dat a ukazatele na data. Požadavek na velikost příchozích dat a požadavek na získání ukazatele na příchozí data je zaslán v kroku „Přečtení příchozích dat po Ethernetu“.

4.7.6 Analýza dat

Příkaz Spínači se skládá ze tří bytů. Rozložení informace v prvním bytu je znázorněno tabulkou 3. Kombinace dvou nejvýznamnějších bitů určuje, zde se má digitální výstup nastavit na žádanou

hodnotu, nebo jestli chce tazatel znát aktuální hodnotu výstupu. Třetí možností je, že tazatel chce zjistit jméno Spínače. Tento dotaz je použit, když je síť skenována při lokalizaci Spínačů.

Tabulka 3: Význam bitů v prvním bytu příkazu Spínači

Bit	7	6	5	4	3	2	1	0
Význam	Příkaz		Číslo výstupu					Set/Puls
Nepoužitý	0	0						
Nastavit výstup	0	1						
Získat hodnotu	1	0						
Získat jméno	1	1						
Číslo výstupu			x	x	x	x	x	
Puls								0
Set								1

Kombinace bitů jedna až pět v binární soustavě udává číslo výstupu, který se má nastavit. Při převodu z dvojkové na desítkovou soustavu je bit 1 nejméně významný bit a bit 5 nejvýznamnější bit. Hodnoty výstupu dva až osm odkazují na digitální výstupy dva až osm mikrokontroleru. Ostatní hodnoty výstupů, až do hodnoty třicet jedna, jsou rezervovány pro vícevýstupové Spínače.

Bit na pozici nula s hodnotou jedna nastaví výstup na žádanou hodnotu. Bit na pozici nula s hodnotou nula sepne výstup na dobu určenou druhým a třetím bytem.

Byt dva a tři jsou použity pro určení délky pulsu při příkazu k pulzu a pro rozlišení zapnutí a vypnutí při příkazu k nastavení výstupu. Byty dva a tři dávají v kombinaci šestnácti bitovou hodnotu. Při kombinování bytů je byte dva významnější a byte tři méně významný. Kombinace bitů udává delku pulsu v milisekundách při příkazu k pulsu. Minimální délka pulsu je tedy nula milisekund a maximální 65,535 sekund.

4.7.7 Odpověď na příchozí data

O odeslání odpovědi se stejně jako o veškerou komunikaci přes Ethernet stará knihovna EtherCard. Odpověď na získání hodnoty výstupu je „HIGH“ v případě, že je výstupní relé sepnuté a „LOW“ v případě, že je výstupní relé rozepnuté. Odpověď na získání jména je jméno Spínače uložené v paměti EEPROM.

4.7.8 Ověření funkčnosti spojení

Spínač má mechanismus na ověření funkčnosti spojení s lokální sítí. Spojení je ověřováno od spuštění hlavní smyčky programu mimo servisní mód. Ověřování funguje tak, že je každých deset sekund odeslána zpráva „Echo request“ pomocí protokolu ICMP na výchozí bránu. Program v hlavní smyčce ověřuje, zda příchozí data nejsou reakcí na ping od výchozí brány. V případě,

že ano, je proměnná „pingOk“ nastavená na hodnotu jedna. Před odesláním každé další zprávy „Echo request“ je ověřen stav proměnné „pingOk“. V případě, že je proměnná ve stavu nula, dojde k pokusu o obnovení spojení. Ověření, zda přichází data obsahující odpověď na „Echo request“, probíhá v kroku „Reakce na přichozí odpověď“. V dalších krocích je obsloužena sériová komunikace, výstupní relé a připojení k místní síti.

4.7.9 Obsluha sériové linky

Sériová linka slouží ke konfiguraci Spínače a jeho diagnostice. O obsluhu sériové linky se stará nativní třída Serial. Ke čtení sériové komunikace a analýze dat dochází tak dlouho, dokud jsou v zásobníku sériové linky data. Při obsluze přichozích dat sériové linky dojde nejprve k vyčtení jednoho bytu ze zásobníku sériové linky. Z bytů je postupně skládán příkaz a program na příkaz reaguje. Formát příkazu popisuje tabulka 4. Vstupní data při nastavování IP adresy a IP adresy implicitní brány jsou čtyři hodnoty 0-255, které korespondují s hodnotou této adresy. Povolení DHCP reaguje pouze na jeden znak následující za příkazem. Tímto znakem může být ASCII znak „0“ pro zákaz DHCP nebo „1“ pro povolení. Ostatní příkazy nevyžadují vstupní data.

Při přijetí žádosti o EEPROM data jsou data z EEPROM zkopírována a odeslána jako odpověď. Odpověď na dotaz na stav spojení je znak „1“ v případě, že Spínač od posledního pokusu o připojení do místní sítě obdržel odpověď na „Echo request“. V opačném případě je odpověď znak „0“. Při dotazu na IP je vrácena IP adresa zařízení. Tento dotaz má význam pouze v případě dynamické IP. Aktivace servisního módu deaktivuje všechny funkce Spínače prováděné v hlavní smyčce kromě obsluhy sériové linky a obsluhy přepínače výstupů, který přestane vykonávat veškerou činnost po skončení všech pulsů. Po přijetí žádosti o restart se zařízení restartuje.

Tabulka 4: Formát sériové komunikace

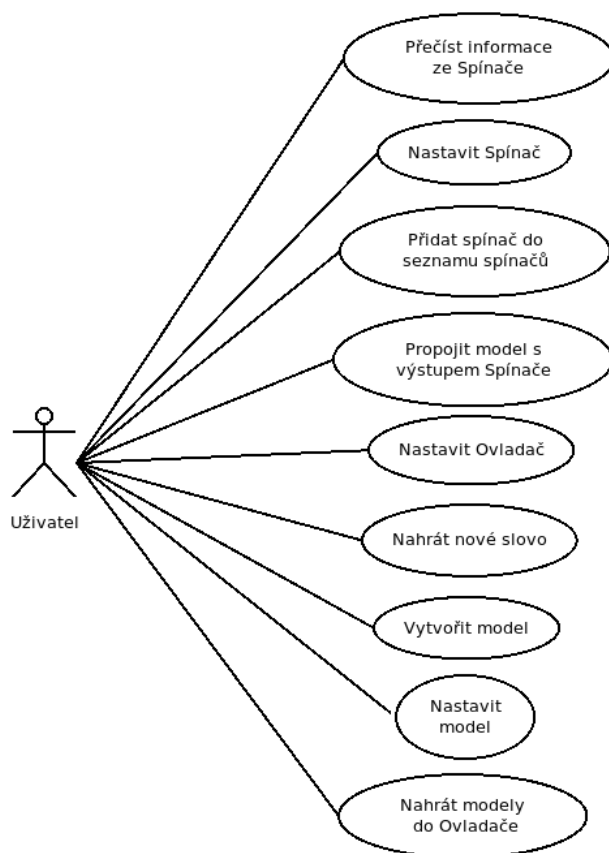
Pořadí znaků	0	1	2	3	4	5
Význam	Úvodní znak	Příkaz	Vstupní data			
Nastavit IP	94	'p'	0 - 255	0 - 255	0 - 255	0 - 255
Povolit DHCP	94	'd'	'0' '1'			
Nastavit IP implicitní brány	94	'g'	0 - 255	0 - 255	0 - 255	0 - 255
Nastavit jméno Spínače	94	'n'	0 - 255	0 - 255	0 - 255	0 - 255
Žádost o EEPROM data	94	'a'				
Žádost o restart	94	'r'				
Aktivace servisního módu	94	's'				
Deaktivace servisního módu	94	'c'				
Dotaz na stav spojení	94	'k'				
Dotaz na IP	94	'i'				

4.7.10 Přepínač výstupů

Přepínač výstupů slouží k obsluze výstupů ovládaných pulsy. Každý výstup aktivovaný pulsem je uložen do pole spolu s časem požadované deaktivace. Přepínač toto pole prochází a kontroluje, zda už nenastal čas na deaktivaci výstupu. V případě, že ano, výstup deaktivuje.

4.8 PC Program

Uživatelský program je napsaný v Qt frameworku v jazyce C++. Všechny funkce, které může uživatel využít, jsou znázorněny diagramem na obrázku 20. Nezkompilovaný program je v příloze A, Uživatelský program - projekt. Zkompilovaný program pro Ubuntu s konfiguračními soubory je v příloze A, Uživatelský program - aplikace.



Obrázek 20: Diagram užití programu na konfiguraci Hlasového ovladače a Spínače

Uživatelský program obsahuje dva prvky stromové struktury a okno se záložkami, viz obrázek 21. Pravá stromová struktura slouží k zobrazení všech uživatelských modelů, citlivosti modelů a jejich typu. Levá stromová struktura slouží k zobrazení uložených nastavení Spínačů. Okno se záložkami obsahuje záložku „Switcher“ a záložku „Record“. Záložka Switcher slouží k nastavení Spínačů, více v sekci 4.8.1. Záložka record slouží ke správě modelů, více v sekci 4.8.6.

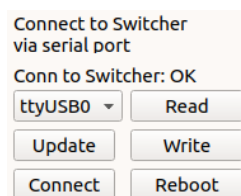
Name	Value	Model	sensitivity	Type
▼ name	EC01	smart_mirror.umd	0.5	Key word
DHCP	true	snowboy.umd	0.5	Key word
IP	192.168.1.111	koupelně.pmdl	0.45	Place
GatewayIP	192.168.1.1	kuchyně.pmdl	0.5	Place
▼ DO1	koupelně.2000	obyvak.pmdl	0.5	Place
▼ DO2	obyvak.pn 0	Pocitaci.pmdl	0.5	Key word
▼ DO3	kuchyně.pn 0	rozsvit.pmdl	0.5	Command
▼ DO4	obyvak.pn 0	zhasni.pmdl	0.5	Command
DO5		svetlo.pmdl	0.54	Place
DO6		alexa.umd	0.5	Key word
DO7		Vypni.pmdl	0.49	Command
▼ name	EC02	vypni2.pmdl	0.6	Command
▼ name	EC03			

Obrázek 21: Pravá stromová struktura nazvaná jako seznam klíčových slov a levá stromová struktura nazvaná jako seznam Spínačů

4.8.1 Spojení se Spínačem

Spínač je možné připojit k počítači pomocí USB, více o HW stránce propojení v sekci 3.2.

Spínač je možné konfigurovat z počítačového programu ze záložky „Switcher“ pomocí konfiguračních tlačítek, viz obrázek 22. Pomocí programu je možné se připojit ke Spínači, přčíst nastavení Spínače, nahrát do spínače novou konfiguraci a uložit konfiguraci. Také je zde možné nastavit chování jednotlivých digitálních výstupů Spínače.



Obrázek 22: Spojení se Spínačem

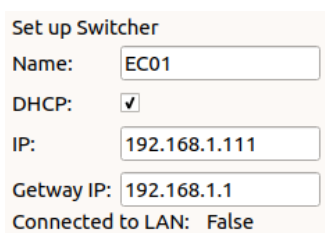
Tlačítko Update slouží k vyčtení všech sériových portů v počítači, i virtuálních, ke kterým je připojeno nějaké zařízení. Tyto porty jsou vepsány do kombinovaného pole nad tlačítkem.

Tlačítko Connect vytvoří sériové spojení se zařízením vybraným pomocí kombinovaného pole nad tlačítkem Update. Po vytvoření spojení se program pokusí přepnout Spínač do servisního módu tím, že periodicky zapisuje „s“ na sériový port. Jakmile dostane domluvenou odpověď tj. „s“ je Spínač přepnutý do servisního módu a status textového pole „Conn to Switcher“ se přepne z „NOK“ na „OK“.

4.8.2 Přečtení konfigurace uložené ve Spínači

Tlačítko Read slouží k přečtení konfigurace Spínače. Po zmáčknutí je vyčištěn zásobník sériové linky a zapsán požadavek na vyčtení konfigurace „^a“. Po odeslání požadavku program čeká na příchozí data, která po přijetí zobrazí v prvcích pod nadpisem „Set up Switcher“, viz obrázek 23. Po přijetí dat se program dotáže požadavkem „^k“, zda se Spínač před přechodem do servisního

módu dokázal připojit k místní síti. Tato informace může pomoci při konfiguraci sítě a je zapsána do textového pole „Connected to LAN“.



Set up Switcher

Name:

DHCP: ☒

IP:

Gateway IP:

Connected to LAN: ☐ False

Obrázek 23: Pole pro konfiguraci Spínače

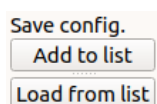
4.8.3 Konfigurace Spínače

Tlačítko Write zapíše konfiguraci z polí pod nadpisem „Set up Switcher“, viz obrázek 23 do paměti Spínače. Parametry jsou zapisovány postupně v pořadí Jméno zařízení, IP adresa Spínače, IP adresa brány a údaj o tom, zda má mít Spínač povolené DHCP. Více o rozložení komunikovaných dat v sekci 4.7.9.

Tlačítkem Reboot je Spínači poslán příkaz na restartování.

4.8.4 Uložení konfigurace Spínače

Tlačítko „Add to list“, viz obrázek 24 slouží k uložení konfigurace do seznamu všech Spínačů. Každý spínač v seznamu musí mít unikátní jméno. Tlačítkem „Load from list“ je možné načíst vybranou konfiguraci ze seznamu všech Spínačů do editovatelných polí. Výběr se provádí v pravé stromové struktuře, viz obrázek 21, kliknutím na položku.



Save config.

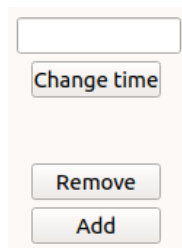
Add to list

Load from list

Obrázek 24: Tlačítka pro uložení a načtení konfigurace

4.8.5 Editace konfigurace Spínače

Po zmáčknutí tlačítka „Change time“, viz obrázek 25, se hodnota v editovatelném poli nad tlačítkem vloží k vybranému klíčovému slovu v levé stromové struktuře v digitálním výstupu Spínače. Časová hodnota reprezentuje dobu v milisekundách, po kterou bude výstup sepnutý. Nula znamená, že výstup zůstane v požadovaném stavu do dalšího příkazu. Spojení slov s digitálním výstupem a jejich časová hodnota se jako jediný z parametrů v záložce Switcher ukládá do Hlasového ovladače.



Obrázek 25: Editace konfigurace

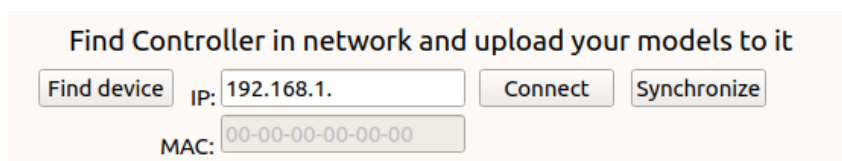
Tlačítko „Remove“ odstraní vybraný model z digitálního výstupu, ke kterému je přiřazen. K přiřazení modelu slouží tlačítko „Add“. Při přiřazování modelu k digitálnímu výstupu musí být model v seznamu modelů vybrán. V seznamu Spínačů musí být zvolen konkrétní digitální výstup, do kterého se model přiřadí.

4.8.6 Vytváření modelů

Záložka Record slouží k vytváření nových modelů. Panel je rozdělen do čtyř částí. První část zhora slouží k vyhledání, ověření připojení a synchronizaci modelů. Pomocí druhé části je možné nahrát příkaz. Pomocí třetí části je z nahraných příkazů vytvořen model. Čtvrtá část slouží ke správě konkrétního modelu.

4.8.7 Připojení k Hlasovému ovladači

Po zmáčknutí tlačítka „Find device“, viz 26, program vyhledá Hlasové ovladače v síti. IP adresa prvního nalezeného Hlasového ovladače je vepsána do pole IP, jeho MAC adresa do pole MAC. IP adresa může být také upravena ručně. Tlačítkem „Connect“ se naváže spojení se zařízením na adrese v poli IP. Tlačítko „Synchronize“ slouží k nahrání modelů a propojovací tabulky modelů do Hlasového ovladače.

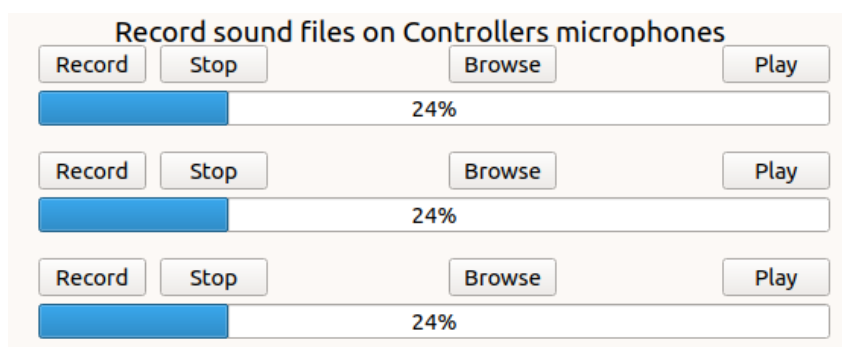


Obrázek 26: Pole pro navázání komunikace s Hlasovým ovladačem

4.8.8 Nahrávání klíčových slov

Úspěšnost správné detekce klíčového slova je vyšší, když jsou podmínky při vytváření modelu a podmínky při detekci podobné. Proto je vhodné klíčová slova pro detekci nahrávat na mikrofonu Hlasového ovladače. K nahrávání slov slouží druhá část záložky, viz obrázek 27. Tlačítko „Record“ spustí nahrávání na Hlasovém ovladači. Tlačítko „Stop“ zastaví nahrávání a stáhne model z Hlasového ovladače do počítače. Tlačítko „Play“ umožní nahrané slova přehrát.

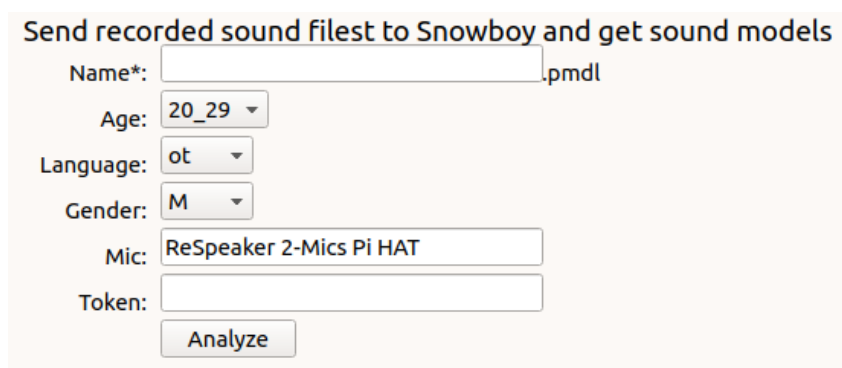
V případě, že uživatel má nahrané klíčové slovo mimo Hlasový ovladač, může ho použít místo slova nahraného Hlasovým ovladačem. K tomuto účelu slouží tlačítka „Browse“.



Obrázek 27: Pole pro nahrávání klíčových slov

4.8.9 Vytváření modelů

Třetí sekce je určena k vytvoření modelu, viz obrázek 28. V této sekci je nutné vyplnit povinné pole se jménem modelu. Pod tímto jménem bude model nadále veden a změna jména není možná. Pole věk, jazyk, pohlaví a mikrofon pomáhají službě na generování modelů vygenerovat kvalitnější model. V případě, že chce uživatel použít jiný než defaultní token pro generování, může ho vyplnit v poli „Token“. Tlačítko „Analyze“ odešle nahraná slova službě Snowboy pomocí dotazu POST, přijme a pojmenuje model, který služba vrátí jako výstup.



Obrázek 28: Pole pro vytváření modelů

4.8.10 Uložení modelu

V této sekci může uživatel načíst model ze seznamu modelů tlačítkem „Load“ nebo přidat do seznamu model, který v seznamu není, tlačítkem „Browse“, viz obrázek 29. Uživatel tímto způsobem může přidat model, který vytvořil, pomocí webové aplikace Snowboy. Načtenému modelu lze změnit citlivost. Čím má model vyšší citlivost, tím snadnější je jeho detekce. Vysoká citlivost může zvýšit počet falešných rozpoznání.[4]

Add sound models to table and adjust sensitivity

Name:

Sensitivity: 50 %

layer:

Obrázek 29: Pole pro uložení modelu

5 Verifikace

Pro potřeby verifikace byl vytvořen speciální program pro modul na detekci slov. Nově vytvořený program je kopií klasického programu ochuzenou o komunikaci s reléovými moduly. Díky absenci detekce modulů při startu programu se zrychlila spouštěcí sekvence programu a díky tomu probíhal test svižněji. Kvůli deaktivaci reakční části programu nastal problém s ověřováním, zda bylo vyslovené slovo detekováno správně. K ověření správnosti detekce využil testující zprávy ladící konzoly. Nezkompilovaný program je v příloze A, Verifikační program - projekt, Zkompilovaný program pro Raspbian v příloze A, Verifikační program - aplikace.

5.1 Forma verifikace

Verifikace se skládala ze čtyř fází. Vytvoření uživatelských modelů řečníky, nastavení citlivosti modelů, testu správnosti detekce při použití modelů řečníka a testu správnosti detekce při použití modelů testujícího.

5.1.1 Průběh vytvoření modelů

Každý řečník si pomocí uživatelského programu vytvořil pět modelů. Modely byly vytvořeny pro slova „Počítači“, „Zhasni“, „Rozsviť“, „V kuchyni“, „V obývací“. Slova byla vybrána s ohledem očekávané použití a jazyk řečníků. Vytvořené modely jsou v příloze A, Modely. Modely jsou rozděleny podle jmen řečníků. Po vytvoření modelů byly modely přesunuty do modulu na detekci slov a nastala fáze nastavení citlivostí.

5.1.2 Průběh nastavení citlivosti modelů

Nastavení citlivosti probíhalo tak, že byl s řečníkem veden rozhovor v blízkosti modulu se zapnutou detekcí. Řečník byl také dle potřeby vyzván k vyslovení některého z detekovaných slov. Citlivost modulů byla upravována testujícím, dokud modul nepřestal detekovat klíčová slova v době, kdy nebyla vyslovena a zároveň při vyslovení klíčového slova toto slovo správně detekoval. Citlivost byla nastavována desítky sekund až jednotky minut v závislosti na podobnosti mezi slovem vysloveným při tvorbě modelů a slovem vysloveným v době nastavování citlivosti. Vytvořené váhové soubory jsou v příloze A, Váhy. Po nastavení citlivosti začal test správnosti detekce při použití modelů řečníka.

5.1.3 Průběh testu správnosti detekce při použití modelů řečníka

Řečník byl nejprve poučen, že modul detekuje slova ve třech fázích. V první fázi čeká na klíčové slovo. V tomto případě je klíčové slovo pouze jedno, a to „Počítači“. Po detekci klíčového slova se LED na modulu rozsvítí modře. Řečník byl poučen, že má klíčové slovo opakovat, dokud se LED nerozsvítí modře. Testující si přitom zapisoval počet pokusů do rozsvícení LED modře.

Po detekci klíčového slova modul očekává příkaz. Příkaz v tomto případě může být „Rozsviť“ nebo „Zhasni“. Po detekci příkazu se LED modulu rozsvítí zeleně. Stejně jako při detekci klíčového slova i v tomto případě byl řečník instruován opakovat slovo místa, dokud se nerozsvítí zelená LED. Také v tomto případě si testující zapisoval počet pokusů do úspěšné detekce.

Po detekci příkazu modul očekává název výstupu na ovládání. V tomto případě „V kuchyni“ nebo „V obývací“ . Po detekci názvu umístění výstupu se barva LED změní na bílou. Řečník byl opět instruován opakovat slovo do úspěšné detekce a testující si i tentokrát zapisoval počet pokusů.

Řečník při každé změně podmínek opakoval sekvence „Počítači - Rozsviť - V kuchyni“, „Počítači - Zhasni - V kuchyni“, „Počítači - Rozsviť - V obývací“ a „Počítači - Zhasni - V obývací“. Řečník opakoval sekvence ze tří vzdáleností při každé změně okolního rušení. Vzdálenosti mezi řečníkem a modulem byly půl metru, jeden metr a dva metry. Pro test byly zvoleny tři typy okolního zvukového rušení. První typ bylo okolí bez přidaného rušení nazváno jako „klid“, druhým typem bylo rušení jedním řečníkem nazvaný jako „hluk 1“ a třetím typem okolního rušení bylo rušení dvěma současně mluvícími řečníky nazvaný jako „hluk 2“.

Testování detekce v prostředí „klid“ probíhalo v místnosti se zavřenými dveřmi i okny. Hladina hluku v místnosti se časem měnila a je proto uvedena u každého testujícího zvlášť. Hladina hluku se pohybovala mezi 30 a 37 dBA.

Testování detekce v prostředí „hluk 1“ probíhalo ve stejné místnosti jako testování v prostředí „klid“. V místnosti byl puštěn reprodukováný čtený projev Lorem Ipsum. Hlasitost čteného projevu byla upravena tak, aby průměrná hladina hluku vedle mikrofonu modulu byla 45 dBA. Čtený projev Lorem Ipsum je v příloze A, Lorem Ipsum.[45]

Testování detekce v prostředí „hluk 2“ probíhalo obdobně jako testování v prostředí „hluk 1“ s tím rozdílem, že reprodukováné čtené projevy byly puštěny dva místo jednoho. Rozdílná byla také průměrná hladina hluku, a to 50 dBA.

5.1.4 Průběh testu správnosti detekce při použití modelů testujícího

Průběh testu správnosti detekce při použití modelů testujícího je obdobný jako průběh testu správnosti detekce při použití modelů řečníka s tím rozdílem, že test probíhá pouze v prostředí „klid“ a vzdálenosti jeden metr. Rozdíl je také v tom, že projev řečníka je porovnáván s modelem testujícího. Tento test testuje úspěšnost detekce příkazu od řečníka, který nemá vlastní model.

5.2 Naměřená data

5.2.1 Naměřená data testu správnosti detekce při použití modelů řečníka

Následující tabulky ukazují naměřená data testu správnosti detekce při použití modelů řečníka 13 řečníků. Mezi řečníky jsou rovnoměrně zastoupena obě pohlaví a všechny věkové kategorie.

Čísla v tabulkách určují číslo pokusu, při kterém modul detekoval správné slovo. Například „1“ znamená, že modul detekoval slovo po jeho prvním vyslovení řečníkem, „3“ znamená, že

systém na vyslovené slovo dvakrát nereagoval a zareagoval až na potřetí. V datech se také vyskytuje hodnota „inf“. Tato hodnota značí, že systém ani po opakovaných pokusech slovo nedetekoval. Další možností je hodnota „-1“. V případě, že je hodnota „-1“, znamená to, že systém detekoval jiné slovo, než bylo ve skutečnosti řečníkem vysloveno.

Tabulka 5: Naměřená data řečníkem Gabriela ml. testu správnosti detekce při použití modelů řečníka

Gabriela ml									
	ticho 30db			hluk 1			hluk 2		
	0,5m	1m	2m	0,5m	1m	2m	0,5m	1m	2m
počítači	1	1	1	2	1	3	1	1	1
rozsviť	1	1	1	1	-1	1	-1	1	-1
v kuchyni	1	1	1	1	1	2	1	1	1
počítači	1	1	1	1	1	1	3	1	1
zhasni	1	1	1	1	2	1	1	1	1
v kuchyni	1	1	1	1	1	2	1	1	1
počítači	1	1	1	1	1	2	2	1	1
rozsviť	1	1	1	1	-1	1	1	-1	2
v obýváku	1	1	1	1	1	1	1	1	1
počítači	2	1	1	1	1	1	1	1	1
zhasni	1	1	1	1	1	1	1	1	1
v obýváku	1	1	1	1	1	1	1	1	1

Tabulka 5 ukazuje naměřená data řečníka Gabriela ml. V tomto případě systém v tichém prostředí detekoval slova správně a většinou napoprvé. V hluku systém nereagoval jen občas. Chybné rozpoznání se nachází pouze u slova „rozsviť“ a pravděpodobně by mohlo být sníženo kvalitnějším nastavením citlivosti.

Tabulka 6: Naměřená data řečníkem Gabriela st. testu správnosti detekce při použití modelů řečníka

Gabriela st									
	ticho 30db			hluk 1			hluk 2		
	0,5m	1m	2m	0,5m	1m	2m	0,5m	1m	2m
počítači	1	1	1	1	1	1	3	1	1
rozsviř	1	1	1	1	1	1	1	1	1
v kuchyni	1	1	1	1	1	1	1	2	2
počítači	1	1	1	1	1	1	2	1	1
zhasni	-1	1	-1	1	-1	-1	1	1	-1
v kuchyni	1	1	1	1	1	1	1	1	2
počítači	1	1	1	1	1	1	3	1	1
rozsviř	1	1	1	1	1	1	1	1	1
v obýváku	1	1	1	1	1	-1	1	-1	1
počítači	1	1	1	1	1	4	1	3	1
zhasni	1	1	-1	1	-1	-1	1	2	-1
v obýváku	1	1	1	1	1	-1	-1	-1	-1

Naměřená data v tabulce 6 ukazují kvalitní detekci systému. Tentokrát měl systém problém se slovy „Zhasni“ a „V obýváku“. Chybovost detekce by pravděpodobně bylo možné také snížit kvalitnějším nastavením citlivosti.

Tabulka 7: Naměřená data řečníkem Hana testu správnosti detekce při použití modelů řečníka

Hana									
	ticho 30db			hluk 1			hluk 2		
	0,5m	1m	2m	0,5m	1m	2m	0,5m	1m	2m
počítači	1	1	1	1	2	1	2	1	5
rozsviř	1	1	1	1	1	-1	1	-1	1
v kuchyni	1	1	1	1	1	1	1	1	3
počítači	1	1	1	1	1	3	1	1	6
zhasni	1	1	1	1	1	2	1	2	1
v kuchyni	1	1	1	1	2	1	1	2	1
počítači	1	1	1	4	3	5	1	2	6
rozsviř	1	1	1	1	-1	1	1	1	-1
v obýváku	1	1	1	2	1	1	1	1	1
počítači	1	1	1	2	4	1	1	1	inf
zhasni	1	1	1	1	1	2	3	1	1
v obýváku	1	1	1	1	1	1	1	1	2

V případě tohoto řečníka (tabulka 7) rostl problém modulu detekovat slovo s rostoucím hlukem a vzdáleností. Chybovost detekce slova „rozsvít“ by stejně jako v minulých případech bylo možné snížit kvalitnějším nastavením citlivosti.

Tabulka 8: Naměřená data řečníkem Iveta testu správnosti detekce při použití modelů řečníka

Iveta									
	ticho 30db			hluk 1			hluk 2		
	0,5m	1m	2m	0,5m	1m	2m	0,5m	1m	2m
počítači	1	4	1	1	2	2	7	1	1
rozsvít	1	4	1	1	-1	1	1	4	2
v kuchyni	1	5	2	1	2	1	1	1	1
počítači	1	1	1	1	1	1	2	1	1
zhasni	1	1	-1	1	2	1	1	1	1
v kuchyni	1	1	1	1	2	1	1	1	1
počítači	1	1	1	5	2	1	1	1	1
rozsvít	1	2	1	-1	1	1	2	5	1
v obýváku	1	1	1	2	3	1	1	1	1
počítači	3	1	1	6	1	1	2	2	1
zhasni	3	1	1	-1	-1	1	1	1	1
v obýváku	3	1	1	4	1	2	1	1	1

V případě řečníka „Iveta“ (tabulka 8) je patrné, že modul měl problém se slovy „rozsvít“ a „zhasni“. Jelikož se jedná o slova ze stejné skupiny, chybovost není možné snížit nastavením citlivosti. V tomto případě by pro zkvalitnění detekce mohlo pomoci pouze vytvoření nových modelů těchto slov.

Tabulka 9: Naměřená data řečníkem Jiří ml. testu správnosti detekce při použití modelů řečníka

Jiří ml									
	ticho 37db			hluk 1			hluk 2		
	0,5m	1m	2m	0,5m	1m	2m	0,5m	1m	2m
počítači	1	1	1	1	1	2	3	2	1
rozsviř	1	1	1	1	1	1	-1	1	1
v kuchyni	1	1	1	1	1	1	1	2	1
počítači	1	1	1	1	1	1	1	1	1
zhasni	1	1	1	1	1	1	1	1	1
v kuchyni	1	1	1	1	1	1	2	1	1
počítači	1	1	1	3	1	1	1	1	1
rozsviř	1	1	1	1	1	1	1	1	1
v obýváku	1	1	1	1	1	1	1	1	1
počítači	1	1	1	1	1	1	3	2	4
zhasni	1	1	1	1	1	1	1	1	1
v obýváku	1	1	-1	1	-1	-1	1	4	1

Tabulka 10: Naměřená data řečníkem Jiří st. testu správnosti detekce při použití modelů řečníka

Jiří st									
	ticho 30db			hluk 1			hluk 2		
	0,5m	1m	2m	0,5m	1	2m	0,5m	1m	2m
počítači	1	1	1	1	1	1	1	1	3
rozsviř	1	1	1	1	1	-1	1	1	1
v kuchyni	1	1	1	1	4	1	1	5	1
počítači	1	1	1	1	1	1	2	1	1
zhasni	1	1	1	1	1	1	1	1	1
v kuchyni	1	3	1	3	3	1	1	1	2
počítači	1	1	1	1	1	1	1	1	1
rozsviř	1	1	1	1	1	-1	1	1	1
v obýváku	1	1	1	1	1	1	1	1	1
počítači	1	1	1	1	1	1	1	1	1
zhasni	1	1	1	1	1	-1	1	-1	1
v obýváku	1	1	1	1	-1	1	1	1	1

Tabulka 11: Naměřená data řečníkem Karel testu správnosti detekce při použití modelů řečníka

Karel									
	ticho 30db			hluk 1			hluk 2		
	0,5m	1m	2m	0,5m	1m	2m	0,5m	1m	2m
počítači	7	1	1	3	inf	inf	inf	inf	inf
rozsviř	1	4	2	inf	inf	inf	inf	inf	inf
v kuchyni	1	1	1	inf	inf	inf	inf	inf	inf
počítači	2	1	1	inf	inf	inf	inf	inf	inf
zhasni	1	1	1	inf	inf	inf	inf	inf	inf
v kuchyni	1	1	1	inf	inf	inf	inf	inf	inf
počítači	1	1	1	inf	inf	inf	inf	inf	inf
rozsviř	2	5	1	inf	inf	inf	inf	inf	inf
v obýváku	1	1	1	inf	inf	inf	inf	inf	inf
počítači	2	2	1	inf	inf	inf	inf	inf	inf
zhasni	1	1	1	inf	inf	inf	inf	inf	inf
v obýváku	1	1	1	inf	inf	inf	inf	inf	inf

V případě řečníka Karel (tabulka 11) bylo nutné nadprůměrně snížit citlivost všech slov, protože při nastavování docházelo k velkému počtu falešných detekcí. Modul poté nebyl schopen detekovat v hlučném prostředí.

Tabulka 12: Naměřená data řečníkem Karolína testu správnosti detekce při použití modelů řečníka

Karolína									
	ticho 35db			hluk 1			hluk 2		
	0,5m	1m	2m	0,5m	1m	2m	0,5m	1m	2m
počítači	inf	1	1	1	6	1	3	1	1
rozsviř	1	1	1	1	1	1	4	5	-1
v kuchyni	1	1	1	1	1	1	1	1	1
počítači	5	1	1	2	2	1	1	1	1
zhasni	1	1	1	1	1	3	1	1	1
v kuchyni	1	1	1	1	1	2	1	1	5
počítači	2	1	1	4	1	1	1	1	1
rozsviř	1	1	1	1	1	1	1	1	3
v obýváku	4	1	3	1	1	1	2	1	2
počítači	inf	1	1	1	2	1	1	1	3
zhasni	1	1	2	1	-1	2	-1	1	10
v obýváku	1	1	1	1	1	1	1	5	3

Tabulka 13: Naměřená data řečníkem Vladislav testu správnosti detekce při použití modelů řečníka

Vladislav									
	ticho 35db			hluk 1			hluk 2		
	0,5m	1m	2m	0,5m	1m	2m	0,5m	1m	2m
počítači	1	1	1	1	1	1	1	1	1
rozsviř	1	1	1	1	1	-1	1	1	1
v kuchyni	2	1	4	1	1	1	2	1	-1
počítači	1	1	1	1	1	1	1	1	1
zhasni	1	1	1	1	1	1	1	1	1
v kuchyni	1	3	2	1	1	1	2	-1	-1
počítači	1	1	1	1	1	3	1	1	6
rozsviř	1	-1	1	-1	1	1	-1	1	-1
v obýváku	1	2	1	1	2	1	-1	1	1
počítači	1	1	1	1	1	1	1	2	1
zhasni	1	1	1	1	1	1	1	-1	1
v obýváku	1	1	1	1	1	1	1	1	2

Tabulka 14: Naměřená data řečníkem Matěj testu správnosti detekce při použití modelů řečníka

Matěj									
	ticho 30db			hluk 1			hluk 2		
	0,5m	1m	2m	0,5m	1m	2m	0,5m	1m	2m
počítači	3	1	1	1	1	1	5	1	1
rozsviř	1	1	1	1	1	2	1	1	2
v kuchyni	1	1	1	1	1	1	1	1	1
počítači	2	1	1	3	1	1	1	1	1
zhasni	1	1	3	1	1	1	1	2	1
v kuchyni	1	1	1	1	1	1	1	1	1
počítači	1	1	1	2	1	1	1	1	1
rozsviř	1	1	1	1	1	2	1	5	1
v obýváku	1	1	1	1	1	1	1	1	1
počítači	3	1	1	3	1	2	1	1	1
zhasni	1	1	1	1	1	1	1	1	-1
v obýváku	1	1	1	1	3	1	1	2	1

Tabulka 15: Naměřená data řečníkem Michal testu správnosti detekce při použití modelů řečníka

Michal									
	ticho 35db			hluk 1			hluk 2		
	0,5m	1m	2m	0,5m	1m	2m	0,5m	1m	2m
počítači	1	1	1	1	2	1	2	3	2
rozsviř	1	2	1	3	4	3	5	2	2
v kuchyni	1	1	1	2	2	1	inf	inf	inf
počítači	1	1	1	2	1	2	4	3	3
zhasni	1	2	1	1	1	1	1	1	1
v kuchyni	2	1	2	2	4	1	1	inf	1
počítači	1	1	1	1	1	1	3	2	10
rozsviř	2	1	1	1	6	1	1	inf	1
v obýváku	3	1	inf	inf	inf	inf	inf	inf	inf
počítači	1	1	1	2	1	4	5	2	2
zhasni	1	1	1	1	1	1	inf	1m	2
v obýváku	2	1	inf	inf	inf	inf	inf	inf	inf

Tabulka 16: Naměřená data řečníkem Sarah testu správnosti detekce při použití modelů řečníka

Sarah									
	ticho 30db			hluk 1			hluk 2		
	0,5m	1m	2m	0,5m	1m	2m	0,5m	1m	2m
počítači	1	1	1	inf	inf	inf	inf	inf	inf
rozsviř	1	1	1	inf	inf	inf	inf	inf	inf
v kuchyni	1	1	1	inf	inf	inf	inf	inf	inf
počítači	1	1	1	inf	inf	inf	inf	inf	inf
zhasni	1	1	1	inf	inf	inf	inf	inf	inf
v kuchyni	1	1	1	inf	inf	inf	inf	inf	inf
počítači	1	1	1	inf	inf	inf	inf	inf	inf
rozsviř	1	1	-1	inf	inf	inf	inf	inf	inf
v obýváku	1	1	1	inf	inf	inf	inf	inf	inf
počítači	1	1	1	inf	inf	inf	inf	inf	inf
zhasni	1	1	4	inf	inf	inf	inf	inf	inf
v obýváku	1	1	1	inf	inf	inf	inf	inf	inf

Tabulka 17: Naměřená data řečníkem Viktor testu správnosti detekce při použití modelů řečníka

Viktor									
	ticho 30db			hluk 1			hluk 2		
	0,5m	1m	2m	0,5m	1m	2m	0,5m	1m	2m
počítači	1	1	1	1	1	1	1	1	1
rozsviř	1	-1	-1	1	1	1	2	1	1
v kuchyni	1	1	1	1	1	6	1	3	1
počítači	1	1	1	1	1	2	1	1	3
zhasni	1	1	1	1	-1	1	1	1	-1
v kuchyni	1	1	1	1	1	1	1	1	-1
počítači	1	1	1	1	1	1	1	1	2
rozsviř	1	1	-1	-1	1	2	-1	1	1
v obýváku	1	1	1	1	1	1	1	-1	1
počítači	1	1	1	1	1	2	1	1	1
zhasni	1	2	1	1	1	1	-1	-1	1
v obýváku	1	1	1	1	1	2	1	1	1

5.2.2 Naměřená data testu správnosti detekce při použití modelů cizího řečníka

Následující tabulka 18 ukazuje naměřená data testu správnosti detekce při použití modelů řečníka Matěj. Tyto modely jsou pro jednotlivé řečníky cizí. Test byl prováděn v tichu krátce po testu správnosti detekce při použití vlastních modelů. Řečník byl při testu vzdálen od modulu jeden metr.

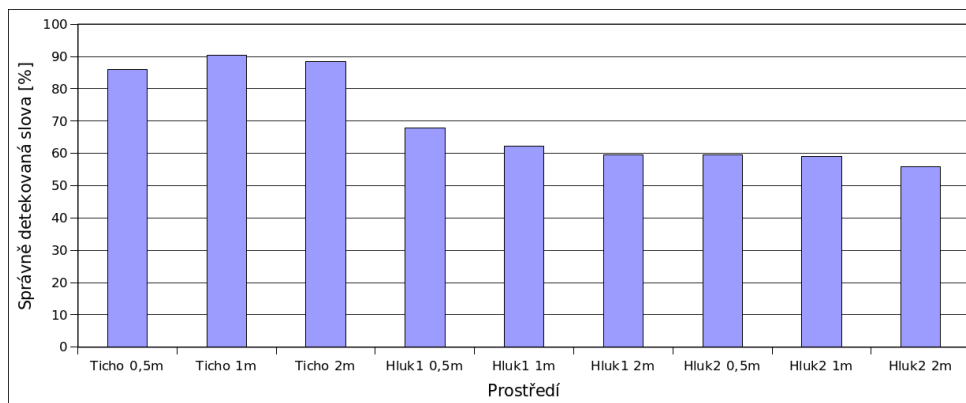
Tabulka 18: Naměřená data všech řečníků testu správnosti detekce při použití modelů cizího řečníka (Matěj) ve sloupcích 2 až 13. Pro srovnání stejný test řečníka Matěj oproti vlastnímu modelu ve sloupci 14

	Gabriela ml	Gabriela st	Hana	Iveta	Jiří ml	Jiří st	Karel	Karolína	Vladislav	Michal	Sarah	Viktor	Matěj
počítači	1	inf	6	1	1	2	1	inf	inf	5	3	1	1
rozsvit	1	-1	1	-1	1	1	1	-1	1	1	1	5	1
v kuchyni	1	1	1	1	1	1	1	1	1	1	3	inf	1
počítači	2	inf	1	2	inf	3	1	inf	inf	1	2	inf	1
zhasni	4	1	1	1	-1	1	1	1	1	1	1	-1	1
v kuchyni	1	1	1	1	1	1	1	1	1	1	1	inf	1
počítači	1	inf	2	2	inf	3	1	inf	inf	1	1	inf	1
rozsvit	1	inf	1	2	1	1	1	inf	1	1	1	inf	1
v obývací	inf	inf	inf	5	-1	1	2	inf	-1	-1	inf	inf	1
počítači	inf	inf	3	3	inf	1	1	inf	inf	1	2	inf	1
zhasni	1	1	1	1	1	1	1	1	1	1	1	inf	1
v obývací	1	1	inf	-1	-1	1	3	1	-1	2	inf	inf	1

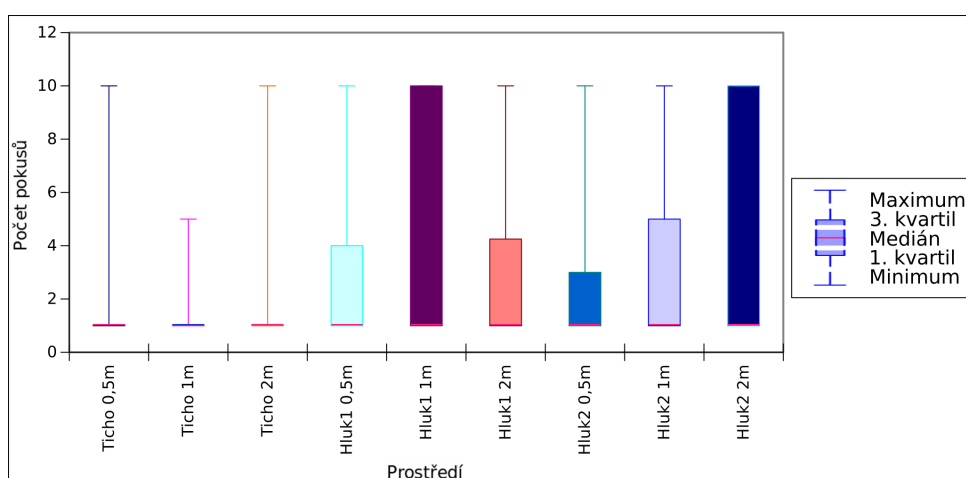
5.3 Zhodnocení výsledků měření

5.3.1 Zhodnocení naměřených dat testu správnosti detekce při použití modelů řečníka

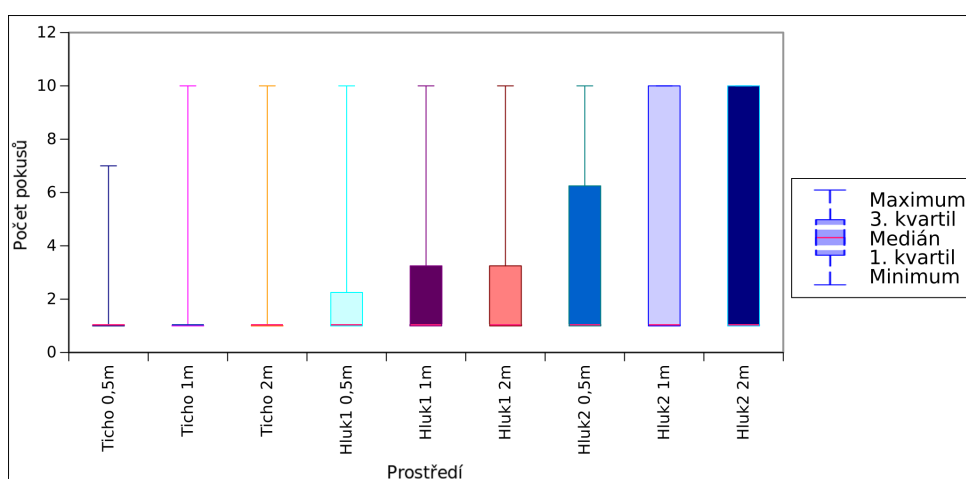
Z celkového počtu 1404 pokusů o detekci bylo napoprvé úspěšných 980, to je 69,8%. Pouze v tichém prostředí bylo správně napoprvé detekovaných 88,25% slov. Výsledky úspěšné detekce napoprvé při různých vzdálenostech a podmínkách jsou znázorněny grafem na obrázku 30. Úspěšnost detekce slova správně napoprvé je 70,06% u žen a 68,54% u mužů. Rozdílnost kvality detekce u různých pohlaví se tedy nepotvrdila. Z výsledků je patrné, že s rostoucím hlukem úspěšnost detekce klesá. Podrobnější výsledky jsou v grafech na obrázcích 31 a 32. V datech pro tyto grafy byl maximální počet pokusů omezen na deset. Chybné rozpoznání slova je bráno jako maximální možný počet pokusů.



Obrázek 30: Úspěšnost správné detekce napoprve při různých vzdálenostech a okolním hluku

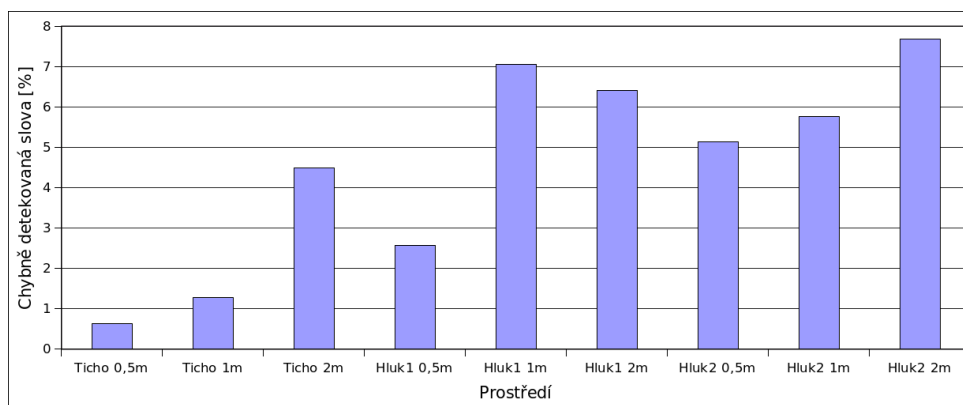


Obrázek 31: Znázornění maximálního a minimálního počtu pokusů, hodnoty prvního a třetího kvartilu a mediánu při různých vzdálenostech a okolním hluku u žen



Obrázek 32: Znázornění maximálního a minimálního počtu pokusů, hodnoty prvního a třetího kvartilu a mediánu při různých vzdálenostech a okolním hluku u mužů

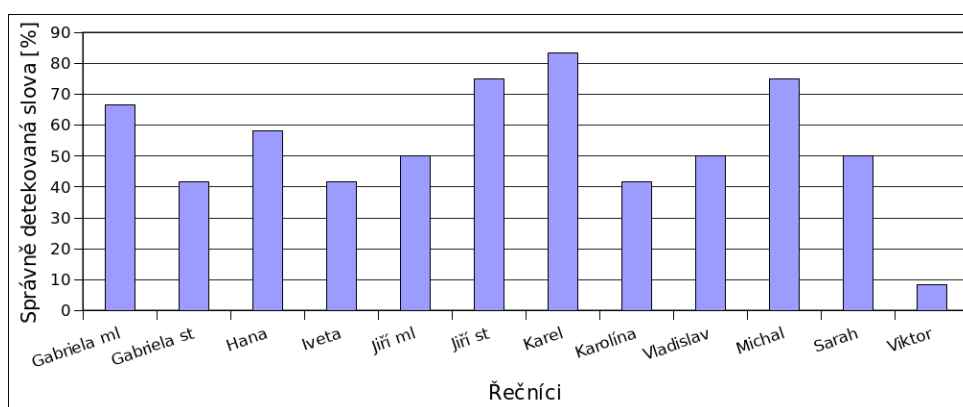
Z celkového počtu bylo slovo chybně detekováno v 4,56% případů. V tichém prostředí bylo slovo chybně detekováno v 2,14% případů a s přibývajícím hlukem a rostoucí vzdáleností řečníka od modulu počet chybně detekovaných slov stoupal. V případě chybně detekovaných slov si systém vedl lépe u mužů s 4% chybovostí než u žen s 5,1% chybovostí. Chybovost detekce při různých vzdálenostech a okolních podmínkách je znázorněna grafem na obrázku 33.



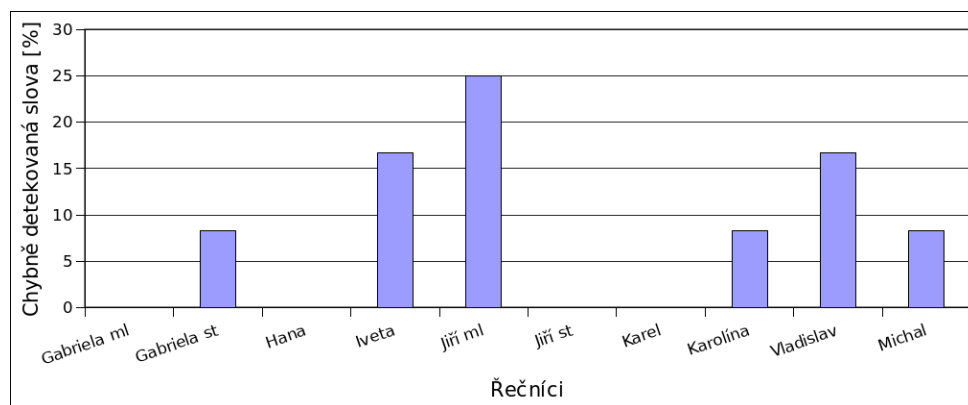
Obrázek 33: Znázornění chybovosti detekce při různých vzdálenostech a okolním hluku

5.3.2 Zhodnocení naměřených dat testu správnosti detekce při použití modelů cizího řečníka

Z celkového počtu 144 pokusů o detekci klíčového slova testu správnosti detekce při použití modelů cizího řečníka bylo 77 slov detekováno správně napoprvé, což je 53,5%. Největším problémem bylo, že systém u některých řečníků nedetekoval některá slova vůbec. Chybovost byla 7,6%, což je více, než chybovost v prostředí hluk 2 u testu správnosti detekce při použití modelů řečníka. Úspěšnost a chybovost detekce u jednotlivých řečníků při použití cizího modelu ukazují grafy na obrázcích 34 a 35.



Obrázek 34: Znázornění úspěšnosti detekce jednotlivých řečníků při použití cizího modelu



Obrázek 35: Znázornění chybovosti detekce jednotlivých řečníků při použití cizího modelu

Cizí model byl vytvořen mužem, úspěšnost detekce správně napoprvé u mužů je 56,9% a u žen 50%. Vliv rozdílného pohlaví řečníka a tvůrce modelu na kvalitu detekce se tedy nepotvrdil.

6 Závěr

Výsledkem práce je funkční systém na detekci slova v mluveném projevu použitelný například v domovní automatizaci. Systém se skládá ze dvou zařízení, z nichž jedno zvuk detekuje a druhé reaguje na detekovaná slova. Modul pro detekci slov k samotné detekci využívá již existující systém Snowboy. Úkolem zde tedy bylo tento systém implementovat do modulu a vytvořit program, který dokáže systém Snowboy zásobovat daty a reagovat na výsledky jeho detekce. Vytvořený program umí také vyhledat reléové moduly v síti a zasílat jim příkazy na sepnutí výstupů. Modul pro detekci slov využívá existující jednodeskový počítač Raspberry Pi 3 a existující modul ReSpeaker. Třetím zařízením v modulu pro detekci slov je audio zesilovač, který byl v rámci práce vyvinut. Stejně tak byla vyvinuta krabička zařízení.

Dalším zařízením, které je součástí systému, je Reléový modul. Deska i program byly vyvinuty v rámci práce. Úkolem reléového modulu je spínat zařízení, která jsou k němu připojena. Reléový modul je konfigurovatelný uživatelskou aplikací.

V rámci práce byla také vyvinuta aplikace, která slouží ke konfiguraci zmíněných dvou zařízení. Pomocí aplikace může uživatel konfigurovat reléový modul a modul pro detekci slov. Aplikací je také možné nahrávat zvuk na modul na detekci slov. Díky této funkci se zvýšila kvalita detekce. Aplikace umí také vytvářet modely pomocí služby Snowboy.

V neposlední řadě byla v rámci práce verifikována kvalita detekce. Testy byly zvoleny tak, aby co nejvíce odrážely běžné použití aplikace. Byla testována kvalita detekce slova různých uživatelů při třech zvukových podmínkách. Systém porovnával projev uživatele s jeho vlastním modelem. Při tomto testu bylo zjištěno, že systém v tichém prostředí správně napoprvé detekuje slovo v 88 %. Také bylo zjištěno, že kvalita detekce klesá s rostoucím okolním hlukem. V práci byla také testována kvalita detekce oproti cizímu modelu. Bylo zjištěno, že správně napoprvé se podařilo slovo detekovat v 53 %. Z výsledku také vyplývá, že kvalita detekce při použití cizího modelu je u různých řečníků rozdílná.

Systém, ač funkční, není dokonalý. Při tvorbě práce jsem vymyslel zdokonalení systému, která budou udávat případný další vývoj. Mohlo by být velice přínosné, kdyby byla vyvinuta zjednodušená verze reléového modulu, která by obsahovala pouze jedno relé a komunikační rozhraní WI-FI místo Ethernetu. Toto zařízení by také mohlo být napájeno přímo z již existujících rozvodů elektřiny ke světlu a umístěno uvnitř světla nebo ve stropě nad ním. Dalším zdokonalením systému by mohlo být vytvoření modulu pro detekci slov na jedné vlastní desce. Deska by mohla obsahovat všechny tři stávající moduly a mohla by být ochuzena o nepoužívané periférie. V neposlední řadě by stála za úvahu úprava krabičky odborníkem přes design.

Literatura

- [1] POWER SUPPLY. In: *Raspberry pi* [online]. Spojené království: The Raspberry Pi Foundation [cit. 2018-01-04]. Dostupné z: <https://www.raspberrypi.org/documentation/hardware/raspberrypi/power/README.md>
- [2] Raspberry Pi. In: *Wikipedia: The Free Encyclopedia* [online]. 14. 12. 2017 [cit. 2018-01-05]. Dostupné z: https://cs.wikipedia.org/wiki/Raspberry_Pi
- [3] Rp2_pinout. In: *Microsoft Docs* [online]. [cit. 2018-01-05]. Dostupné z: https://docs.microsoft.com/en-us/windows/iot-core/media/pinmappingsrpi/rp2_pinout.png
- [4] YAO Xuchen. Snowboy Hotword Detection. In: *GitHub* [online]. San Francisco: GitHub, 2018, 19.2.2018 [cit. 2019-02-02]. Dostupné z: <https://github.com/Kitt-AI/snowboy>
- [5] Snowboy, a Customizable Hotword Detection Engine. In: *Kitt* [online]. Seattle: KITT, 2017, 24. 3. 2017 [cit. 2019-02-03]. Dostupné z: <http://docs.kitt.ai/snowboy/>
- [6] Zvuk. Wikipedie: Otevřená encyklopedie [online]. 2019, 9. 3. 2019 [cit. 2019-03-11]. Dostupné z: <https://cs.wikipedia.org/w/index.php?title=Zvuk&oldid=17028798>
- [7] LAZÁRKOVÁ, Dita. Zvuk a zvukové vlnění. WikiSofia [online]. 2015, 21. 1. 2015 [cit. 2019-03-11]. Dostupné z: https://wikisofia.cz/wiki/Zvuk_a_zvukov%C3%A9_vln%C4%9Bn%C3%AD
- [8] Zvuk. Wikipedie: Otevřená encyklopedie [online]. 2019, 20. 1. 2019 [cit. 2019-03-11]. Dostupné z: <https://cs.wikipedia.org/w/index.php?title=Decibel&oldid=16874918>
- [9] ŠVEC, Jan: Studium mechanicko-akustických vlastností lidského hlasu. (Dizertační práce). [online]. 1996 [cit. 2019-03-11]. Dostupné z: https://www.researchgate.net/publication/311087009_Studium_mechanicko-akustickyh_vlastnosti_zdroje_lidskeho_hlasu_Dizertacni_prace
- [10] Zvuk. Wikipedie: Otevřená encyklopedie [online]. 2019, 20. 1. 2019 [cit. 2019-03-11]. Dostupné z: <https://cs.wikipedia.org/w/index.php?title=Hlas&oldid=16718189>
- [11] Wikipedia contributors. Microphone. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2019, 26. 2. 2019 [cit. 2019-03-13]. Dostupné z: <https://en.wikipedia.org/w/index.php?title=Microphone&oldid=885220518>
- [12] Mikrofon. Wikipedie: Otevřená encyklopedie [online]. 2018, 20. 11. 2018 [cit. 2019-03-12]. Dostupné z: <https://cs.wikipedia.org/w/index.php?title=Mikrofon&oldid=16701404>

- [13] MELLOR, David. 7 important microphone types that you should know and the benefits of each. The Audio Masterclass Newsletter. 2016, Dostupné z: <https://www.audiomasterclass.com/newsletter/7-important-microphone-types-that-you-should-know-and-the-benefits-of-each>
- [14] Electret microphone. Wikipedia, The Free Encyclopedia. [online]. 2018, 13. 6. 2018 [cit. 2019-03-12]. Dostupné z: https://en.wikipedia.org/w/index.php?title=Electret_microphone&oldid=845757445
- [15] Digitalizace signálu. Wikipedie: Otevřená encyklopedie. [online]. 2017, 5. 10. 2017 [cit. 2019-03-17]. Dostupné z: https://cs.wikipedia.org/w/index.php?title=Digitalizace_sign%C3%A1lu&oldid=15424013
- [16] Digitization. Wikipedia, The Free Encyclopedia. [online]. 2019, 5. 3. 2019 [cit. 2019-03-17]. Dostupné z: <https://en.wikipedia.org/w/index.php?title=Digitization&oldid=886298445>
- [17] Kvantování (signál). Wikipedie: Otevřená encyklopedie. [online]. 2019, 3. 1. 2019 [cit. 2019-03-17]. Dostupné z: [https://cs.wikipedia.org/w/index.php?title=Kvantov%C3%A1n%C3%AD_\(sign%C3%A1l\)&oldid=16827754](https://cs.wikipedia.org/w/index.php?title=Kvantov%C3%A1n%C3%AD_(sign%C3%A1l)&oldid=16827754)
- [18] Quantization (signal processing). Wikipedia, The Free Encyclopedia. [online]. 2019, 8. 3. 2019 [cit. 2019-03-17]. Dostupné z: [https://en.wikipedia.org/w/index.php?title=Quantization_\(signal_processing\)&oldid=886796162](https://en.wikipedia.org/w/index.php?title=Quantization_(signal_processing)&oldid=886796162)
- [19] Vzorkování. Wikipedie: Otevřená encyklopedie. [online]. 2018, 7. 6. 2018 [cit. 2019-03-17]. Dostupné z: <https://cs.wikipedia.org/w/index.php?title=Vzorkov%C3%A1n%C3%AD&oldid=16143352>
- [20] Sampling (signal processing). Wikipedia, The Free Encyclopedia. [online]. 2019, 25. 2. 2019 [cit. 2019-03-17]. Dostupné z: [https://en.wikipedia.org/w/index.php?title=Sampling_\(signal_processing\)&oldid=885036021](https://en.wikipedia.org/w/index.php?title=Sampling_(signal_processing)&oldid=885036021)
- [21] Vzorkovací frekvence. Wikipedie: Otevřená encyklopedie. [online]. 2018, 7. 9. 2018 [cit. 2019-03-17]. Dostupné z: https://cs.wikipedia.org/w/index.php?title=Vzorkovac%C3%AD_frekvence&oldid=16429291

- [22] VŠETIČKA, Martin. Kvantování signálu. Encyklopedie fyziky. [online]. 2011, 27. 12. 2011 [cit. 2019-03-17]. Dostupné z: <http://fyzika.jreichl.com/main.article/view/1357-kvantovanisignalu>
- [23] A/D převodník. Wikipedie: Otevřená encyklopedie. [online]. 2018, 12. 12. 2018 [cit. 2019-03-17]. Dostupné z: https://cs.wikipedia.org/w/index.php?title=A/D_p%C5%99evodn%C3%ADk&oldid=16773803
- [24] Analog-to-digital converter. Wikipedia, The Free Encyclopedia. [online]. 2019, 4. 3. 2019 [cit. 2019-03-17]. Dostupné z: https://en.wikipedia.org/w/index.php?title=Analog-to-digital_converter&oldid=886079537
- [25] KESTER, Walt. Which ADC Architecture Is Right for Your Application?. AnalogDialogue. [online]. 2015, 6.2005 [cit. 2019-03-17]. Dostupné z: <https://www.analog.com/en/analog-dialogue/articles/therightadc-architecture.html>
- [26] Sample and hold. Wikipedia, The Free Encyclopedia. [online]. 2018, 7. 6. 2018 [cit. 2019-03-17]. Dostupné z: https://en.wikipedia.org/w/index.php?title=Sample_and_hold&oldid=844825637
- [27] FRENZEL, Lou. What's the Difference Between SAR and Delta-Sigma ADCs?. ElectronicDesign: sn informa business. [online]. 2016, 4. 5. 2016 [cit. 2019-03-17]. Dostupné z: <https://www.electronicdesign.com/adc/whatsdifferencebetweensaranddelta-sigmaadcs>
- [28] VESELÝ, Martin. A/D převodník typu Sigma-Delta. In: SPŠ Elektro a Výpočetka: Elaboráty, ročníkovky atd. [online]. Praha, 2006, 18. 11. 2006 [cit. 2019-03-17]. Dostupné z: www.vesielt.wz.cz/other/sigma_delta_prevodnik.doc
- [29] Broadcom [online katalogový list]. BCM2835 ARM Peripherals. ©2012 [cit. 2019-04-13]. Dostupné z: <https://www.raspberrypi.org/app/uploads/2012/02/BCM2835-ARM-Peripherals.pdf>
- [30] The difference between speaker-dependent and speaker-independent recognition software. SPEECH ANGEL [online]. Switzerland, 2016, 4. 5. 2016 [cit. 2019-04-23]. Dostupné z: <https://speechangel.com/2016/05/04/differencespeakerdependent-speakerindependentrecognitionsoftware/>
- [31] ROUSE, Margaret. Speech recognition. Search Customer Experience [online]. prosinec 2016 [cit. 2019-04-24]. Dostupné z: <https://searchcustomerexperience.techtarget.com/definition/speech-recognition>

- [32] HUNT, Andrew. Q6.1: What is speech recognition?. Comp.speech [online]. 18. 6. 1996 [cit. 2019-04-24]. Dostupné z: <http://www.speech.cs.cmu.edu/comp.speech/Section6/Q6.1.html>
- [33] Kurzekar, Pratik & Deshmukh, Ratnadeep & Waghmare, Dr. Vishal & P Shrishrimal, Pukhraj. (2014). CONTINUOUS SPEECH RECOGNITION SYSTEM: A REVIEW. Asian Journal Computer Science & Information Technology. 4. 62-66. 10.15520/ajcsit.v4i6.3.
- [34] Keyword spotting. Wikipedia, The Free Encyclopedia. [online]. 2019, 7. 3. 2019 [cit. 2019-04-24]. Dostupné z: https://en.wikipedia.org/w/index.php?title=Keyword_spotting&oldid=886697475
- [35] Google Assistant API Terms of Service. Google Assistant SDK [online]. 8. 8. 2018 [cit. 2019-04-24]. Dostupné z: <https://developers.google.com/assistant/sdk/termsofservice>
- [36] Google Assistant. Wikipedia, The Free Encyclopedia. [online]. 2019, 23. 4. 2019 [cit. 2019-04-24]. Dostupné z: https://en.wikipedia.org/w/index.php?title=Google_Assistant&oldid=893762671
- [37] Alexa for Business Requirements. Amazon Alexa [online]. [cit. 2019-04-24]. Dostupné z: <https://developer.amazon.com/docs/alexavoiceservice/alexaforbusiness-requirements.html#alexaforbusiness>
- [38] Security Best Practices. Amazon Alexa [online]. [cit. 2019-04-24]. Dostupné z: <https://developer.amazon.com/docs/alexavoiceservice/securitybest-practices.html>
- [39] Amazon Developer Services Agreement. Amazon Developer [online]. 12. 2. 2019 [cit. 2019-04-24]. Dostupné z: <https://developer.amazon.com/support/legal/da>
- [40] Build Smart Devices with the Alexa Connect Kit (ACK). Amazon Alexa [online]. [cit. 2019-04-24]. Dostupné z: <https://developer.amazon.com/alexa/connecteddevices/alexa-connectkit>
- [41] Alexa Connected Devices. Amazon Alexa [online]. [cit. 2019-04-24]. Dostupné z: <https://developer.amazon.com/alexa/connecteddevices>
- [42] Amazon Alexa. Wikipedia, The Free Encyclopedia. [online]. 2019, 22. 4. 2019 [cit. 2019-04-24]. Dostupné z: https://en.wikipedia.org/w/index.php?title=Amazon_Alexa&oldid=893531833
- [43] BISHOPH. Sopare precision and accuracy. Home of bishoph: Smarthome, speech recognition, voice control, Raspberry Pi and more [online]. 18. 6. 2018 [cit. 2019-04-24]. Dostupné z: <https://www.bishoph.org/sopareprecisionandaccuracy/>

- [44] BISHOPH. Step by step: Raspberry Pi offline voice recognition with SOPARE. Home of bishoph: Smarthome, speech recognition, voice control, Raspberry Pi and more [online]. 18. 6. 2018 [cit. 2019-04-24]. Dostupné z: <https://www.bishoph.org/stepbystepaspberry-piofflinevoicerecognitionwithsopare/#more464>
- [45] 1 hour of lorem ipsum (Latin speech). In: Youtube [online]. 22. 1. 2019 [cit. 2019-04-24]. Dostupné z: <https://www.youtube.com/watch?v=BggrpKfqh1c>. Kanál uživatele Andriy Vasylenko 2.
- [46] KANETKAR, Salil, David HARRISON a Manfred K WARMUTH, KRISHNA GOUDA, Sanjay, ed. Speech Recognition: Keyword Spotting Through Image Recognition. In: Eprint arXiv: 1803.03759 [online]. Březen 2018 [cit. 2019-04-27]. Dostupné z: <https://arxiv.org/pdf/1803.03759.pdf>
- [47] Convolutional neural network. Wikipedia, The Free Encyclopedia. [online]. 2019, 21. 4. 2019 [cit. 2019-04-27]. Dostupné z: https://en.wikipedia.org/w/index.php?title=Convolutional_neural_network&oldid=893419085
- [48] SAINATH, Tara N., Carolina PARADA. Convolutional Neural Networks for Small-footprint Keyword Spotting. In: Google, Inc. New York, NY, U.S.A [online]. 2015 [cit. 2019-04-27]. Dostupné z: https://www.isca-speech.org/archive/interspeech_2015/papers/i15_1478.pdf
- [49] PEDDINTI, Vijayaditya, Daniel POVEY a Sanjeev KHUDANPUR. A time delay neural network architecture for efficient modeling of longtemporal contexts [online]. In: . Center for Language and Speech Processing & Human Language Technology Center of Excellence. Johns Hopkins University, Baltimore, MD 21218, USA [online]. 28. března 2015 [cit. 2019-04-28]. Dostupné z: speak.clsp.jhu.edu/uploads/publications/papers/1048_pdf.pdf
- [50] Recurrent neural network. Wikipedia, The Free Encyclopedia. [online]. 2019, 28. 3. 2019 [cit. 2019-04-28]. Dostupné z: https://en.wikipedia.org/w/index.php?title=Recurrent_neural_network&oldid=889817533
- [51] Time delay neural network. Wikipedia, The Free Encyclopedia. [online]. 2019, 10. 3. 2019 [cit. 2019-04-28]. Dostupné z: https://en.wikipedia.org/w/index.php?title=Time_delay_neural_network&oldid=887118220
- [52] KANETKAR, Salil, David HARRISON a Manfred K WARMUTH, KRISHNA GOUDA, Sanjay, ed. Efficient keyword spotting using time delay neural networks.

In: Eprint arXiv: 1807.04353v2 [online]. 28. 8. 2018 [cit. 2019-04-28]. Dostupné z: <https://arxiv.org/pdf/1807.04353.pdf>

- [53] XIANGANG, Li, Wu XIHONG, ed. CONSTRUCTING LONG SHORT-TERM MEMORY BASED DEEP RECURRENT NEURALNETWORKS FOR LARGE VOCABULARY SPEECH RECOGNITION. In: Eprint arXiv: 1410.4281v2 [online]. 11. 5. 2015 [cit. 2019-04-28]. Dostupné z: <https://arxiv.org/pdf/1410.4281.pdf>
- [54] Zhang, Hongwei & Bao, Feilong & Gao, Guanglai & Zhang, Hui., Comparison on Neural Network based acoustic model in Mongolian speech recognition. 10.1109/IALP.2016.7875921 [online]. listopad 2015 [cit. 2019-04-28]. Dostupné z: https://www.researchgate.net/publication/315513471_Comparison_on_Neural_Network_based_acoustic_model_in_Mongolian_speech_recognition

7 Seznam příloh

- Příloha A: CD

A CD

1. Modely
2. Váhy
3. Program na detekci slova - aplikace
4. Program na detekci slova - projekt
5. Program Spínače - projekt
6. Model Krabičky
7. Zesilovač
8. Reléový modul
9. Uživatelský program - aplikace
10. Uživatelský program - projekt
11. Verifikační program - aplikace
12. Verifikační program - projekt
13. Lorem Ipsum